



Universidad Carlos III de Madrid

Departamento de Tecnología Electrónica

Trabajo de Fin de Grado:

Implementación de un sistema de transmisión y detección de señales M-PAM sobre fibra óptica de plástico

Autor:

Álvaro José Triviño Romero

Grado en Ingeniería Electrónica Industrial y Automática

Supervisado por:

Plinio Jesús Pinzón Castillo

TÍTULO: *Implementación de un sistema de transmisión y detección de señales M-PAM sobre fibra óptica de plástico*

AUTOR: *Álvaro José Triviño Romero*

TUTOR: *Plinio Jesús Pinzón Castillo*

La defensa del presente Proyecto Fin de Carrera se realizó el día 4 de octubre de 2016 siendo calificada por el siguiente tribunal:

PRESIDENTE: Rodríguez Urbano, Francisco José

SECRETARIO: Cabana Garcerán del Val, Elisa

VOCAL: Martín Mateos, Pedro

Habiendo obtenido la siguiente calificación:

CALIFICACIÓN:

Presidente

Secretario

Vocal

Agradecimientos

En primer lugar, me gustaría darle las gracias a mi tutor Jesús ya que sin él este Trabajo de Fin de Grado no hubiera sido posible. Me gustaría destacar tu optimismo y perseverancia en todo lo que he ido consiguiendo, que me ha servido y servirá de motivación para esta y otras metas que me proponga. Muchas gracias por tu dedicación y consejos, ha sido un placer compartir esta etapa tan importante contigo. Tampoco puedo olvidarme de Carmen Vázquez, que fue quien me puso en contacto con Jesús y, por lo tanto, me dio esta oportunidad.

En segundo lugar, me gustaría darle las gracias a mi familia, es especial a mis padres, Juani y José Antonio que desde el minuto cero han estado apoyándome y que siempre han depositado en mí su confianza. Nunca podré agradecerlos todo lo que habéis hecho por mí.

En tercer lugar, tengo que agradecerle a mi jefe Antonio todo su apoyo y facilidades que me ha dado para compaginar la realización de este TFG con el trabajo.

Y por último, a mis amigos, especialmente a Sandra, que siempre ha estado encima de mí, motivándome y haciendo que consiguiera sacar lo mejor de mí mismo, incluso estando a miles de kilómetros.

Resumen

En este documento se ha querido implementar un sistema que permita generar una señal digital modulada por amplitud de pulsos (M-PAM) y transmitirla a través de un enlace de fibra óptica de plástico de salto de índice (SI-POF), para su posterior detección y análisis de calidad, en términos de la tasa de error de bit (BER).

Después de realizar un análisis de la SI-POF (por sus siglas en inglés *Step Index – Plastic Optic Fiber*), se diseñó un sistema de transmisión, cuya generación, modulación, demodulación, codificación, decodificación se realiza mediante SIMULINK y la transmisión de datos a través de una tarjetas de National Instruments.

Debido a la limitación en cuanto a velocidad impuesta por dichas tarjetas, se diseñó un segundo sistema en MATLAB en el cual, una vez generados los datos, se modifican, se codifican y se introducen de un generador de señales arbitrarias (de hasta 500 MHz). Mediante SI-POF se transmite hasta un osciloscopio. Una vez que la transmisión se realiza, se recuperan los datos del osciloscopio y se decodifican y demodulan mediante MATLAB. Finalmente se comparan los datos recibidos con los entregados para analizar la calidad de la transmisión.

Contenido

Capítulo 1: Introducción.....	10
1.1. Objetivos Generales	10
1.2. Fibras ópticas	11
1.3. Estado del arte	18
1.4. Objetivos específicos y alcance del proyecto	19
Capítulo II: Planificación.....	20
2.1. Diagrama de Gantt	21
Capítulo III: Simulaciones y procedimientos.....	22
3.1. Diagrama de bloques general	22
Generación de la señal de transmisión.....	22
Implementación en Simulink	26
3.2. Sistema de transmisión y detección de señales M-PAM propuesto	28
Esquema.....	28
Descripción detallada de componentes	29
Procedimiento de configuración.....	30
Capítulo IV: Resultados experimentales.....	38
4.1. Caso de estudio 1: Portadora a 100 MHz (x6)	38
4.2. Caso de estudio 2: Portadora a 125 MHz (x5)	42
4.3. Caso de estudio 3: Portadora a 167 MHz (x4)	46
4.4. Caso de estudio 4: Portadora a 250 MHz (x3)	50
Capítulo V: Conclusión y posibles mejoras y trabajos futuros.....	53
Anexo A: Presupuesto.....	54
2.1. Costes de software y equipos de medida.....	54
2.2. Trabajo de ingeniería.....	55
2.3. Coste total	55
Anexo B: Código figuras Matlab.....	56
Anexo C: Referencias.....	58

Lista de figuras

Figura 1.1 Tipos de fibras ópticas	11
Figura 1.2 Composición fibra óptica.....	12
Figura 1.3 Modulación DMT.....	15
Figura 1.4 Modulación PAM.....	15
Figura 2.1 Diagrama de Gantt.....	21
Figura 3.1 BER vs Eb/No (dB)	25
Figura 3.2 Esquema sistema de transmisión con tarjetas NI	26
Figura 3.3 Sistema completo	29
Figura 3.4 Software generador de ondas.....	29
Figura 3.5 Generador de ondas	30
Figura 3.6 Osciloscopio	30
Figura 4.1 Datos de transmisión y señal recibida x6. La Línea sólida corresponde a Tx y la punteada a Rx. ..	39
Figura 4.2 Desplazamiento de la señal recibida: correlación x6- La Línea sólida corresponde a Tx y la punteada a Rx.....	39
Figura 4.3 Eliminación de ceros x6. La Línea sólida corresponde a Tx y la punteada a Rx.	40
Figura 4.4 Submuestreo media ponderada x6. La Línea sólida corresponde a Tx y la punteada a Rx.	40
Figura 4.5 BER media ponderada x6.....	41
Figura 4.6 Submuestreo interpolación x6. La Línea sólida corresponde a Tx y la punteada a Rx.	41
Figura 4.7 BER interpolación x6	41
Figura 4.8 Datos de transmisión y señal recibida x5. La Línea sólida corresponde a Tx y la punteada a Rx. ..	43
Figura 4.9 Desplazamiento de la señal recibida: correlación x6. La Línea sólida corresponde a Tx y la punteada a Rx.....	43
Figura 4.10 Eliminación de ceros x5. La Línea sólida corresponde a Tx y la punteada a Rx.	44
Figura 4.11 Submuestreo media ponderada x5. La Línea sólida corresponde a Tx y la punteada a Rx.	44
Figura 4.12 BER media ponderada x5.....	45
Figura 4.13 Submuestreo interpolación x5. La Línea sólida corresponde a Tx y la punteada a Rx.	45
Figura 4.14 BER interpolación x5	45
Figura 4.15 Datos de transmisión y señal recibida x4. La Línea sólida corresponde a Tx y la punteada a Rx.	47
Figura 4.16 Desplazamiento de la señal recibida: correlación x4. La Línea sólida corresponde a Tx y la punteada a Rx.....	47
Figura 4.17 Eliminación de ceros x4. La Línea sólida corresponde a Tx y la punteada a Rx.	48
Figura 4.18 Submuestreo media ponderada x4. La Línea sólida corresponde a Tx y la punteada a Rx.	48
Figura 4.19 BER media ponderada x4.....	49
Figura 4.20 Submuestreo interpolación x4. La Línea sólida corresponde a Tx y la punteada a Rx.	49
Figura 4.21 BER interpolación x4	49
Figura 4.22 Datos de transmisión y señal recibida x3. La Línea sólida corresponde a Tx y la punteada a Rx.	51
Figura 4.23 Desplazamiento de la señal recibida: correlación x3. La Línea sólida corresponde a Tx y la punteada a Rx.....	51

Lista de tablas

Tabla 1.1 Últimas publicaciones sobre SI-POF.....	13
Tabla 1.2 Equipos comerciales BERT	16
Tabla 1.3 Sistemas con una longitud de onda	18
Tabla 1.4 Sistemas con varias longitudes de onda.....	19
Tabla 5 Coste de software y equipos de medida	54
Tabla 6 Coste del trabajo de ingeniería	55
Tabla 7 Coste total del TFG.....	55

Capítulo 1: Introducción

En este capítulo trata sobre los sistemas de comunicaciones en fibra de plástico. Concretamente, se va a tratar la fibra de plástico de salto de índice, que es la empleada en este proyecto, así como sus aplicaciones en el mundo actual.

1.1. Objetivos Generales

Se trata de un proyecto multidisciplinar que combina fotónica, comunicaciones y circuitos optoelectrónicos. Para su realización se requiere soltura para el trabajo de laboratorio, nociones de comunicaciones, manejo de componentes optoelectrónicos y conocimiento del software LabVIEW y MATLAB.

Los objetivos concretos de este proyecto son:

- Mostrar iniciativa y capacidad de integración y de dar solución a problemas técnicos en un sistema de comunicaciones a partir de dispositivos comerciales.
- Realizar un estado del arte acerca de las aplicaciones de fibras POF y de las máximas velocidades de transmisión alcanzadas.
- Experimentar con varios métodos de realizar la transmisión de los datos, con el fin de elegir el más adecuado.
- Estudiar sistemas de transmisión con diferentes velocidades y anchos de bit.
- Comprobar las limitaciones de los equipos empleados durante la experimentación.
- Realizar una guía práctica para la evaluación del BER en sistemas M-PAM sobre SI-POF.

1.2. Fibras ópticas

La fibra óptica es un medio de transmisión de vidrio o plástico en el que los datos se envían en forma de pulsos de luz. Se usa frecuentemente en telecomunicaciones, ya que su velocidad de transmisión es superior a la de un cable de cobre y se pueden enviar grandes cantidades de datos en distancias largas.

El principio de las comunicaciones ópticas es transmitir una señal a través de una fibra óptica a un receptor situado a una distancia determinada. La señal eléctrica se convierte al dominio óptico en el transmisor, y se vuelve a convertir en la señal eléctrica original en el receptor.

La fibra óptica se puede clasificar en diferentes tipos (multimodo o monomodo), dependiendo de cómo viaje la luz a través de la misma. El tipo de fibra está estrechamente relacionada con el diámetro del núcleo y el revestimiento.

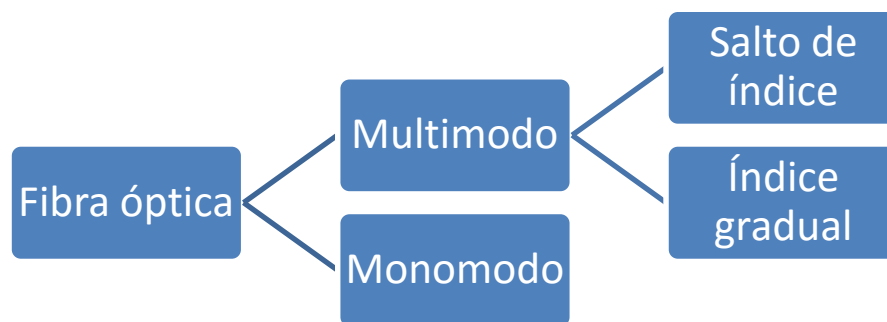


Figura 1.1 Tipos de fibras ópticas

Uno de los tipos de fibra óptica es la que está fabricada por un polímero de plástico. El núcleo de la fibra mide 1 mm aproximadamente y tiene una gran apertura numérica (NA 0.3-0.5). Debido a este gran diámetro, la POF resiste muy bien los desalineamientos así como fuertes vibraciones [1]. Sus principales ventajas respecto a la de vidrio son su precio así como su robustez ante curvaturas pronunciadas [2, 3], por lo que se usa frecuentemente para aplicaciones en distancias cortas para aparatos electrónicos y en vehículos a motor, debido al MOST (Media Oriented Systems Transport) [1]. La primera empresa que introdujo la POF en el mundo automovilístico fue BMW en 2001 [4]. Debido a su fácil manejo y las herramientas básicas con las que se puede manejar, es muy usada en el mundo FTTH (Fiber To The Home) [5, 6, 7]. Otros dos sectores en los que se utiliza este tipo de fibra son la aviación y el sector médico. Ambos sectores tienen una cosa en común, la necesidad de sistemas de comunicación de alta velocidad [4].

La fibra de salto de índice se llama así debido al perfil en forma de escalón del índice de refracción entre el núcleo y el revestimiento de la fibra. Hay dos tipos de sistemas de transmisión con este tipo de

fibra: con una sola longitud de onda y con múltiples longitudes de onda. Estos últimos, buscan una mejora en la capacidad de transmisión.



Figura 1.2 Composición fibra óptica

En la siguiente tabla [8], se pueden observar datos de diferentes publicaciones sobre la SI-POF.

Año	Fabricante	Producto	Diámetro Núcleo [μm]	Atenuación [dB/km]	Longitud de onda [nm]	NA	Comentarios
1963	Du Pont	CROFON	-	1000	650	St.	Primera POF
1964	Du Pont		-	500	650	St.	
1968	Du Pont		-	500	650	St.	Primer SI-POF
1976	Mitsubishi	Eska	-	300	650	St.	
1978	Mitsubishi	Super Eska	-	300	650	St.	
1982	NTT		-	55	568	St.	
1983	Mitsubishi	Eska Extra	-	124	650	St.	4 MHz·km
1983	Mitsubishi	Eska Extra	-	65	570	St.	
1983	Mitsubishi		1000	110	570	St.	
1984	Mitsubishi	Eska Extra	-	150	650	St.	
1985	Asahi		-	80	570	St.	
1991	Mitsubishi	Eska Extra	-	125	650	St.	Hasta 85 °C
1991	Mitsubishi	Eska Extra	-	65	570	St.	
1991	Hoechst		1000	130	650	St.	
1992	Asahi	Luminous-F	-	175	660	0.5	310 MHz·10m AN,LED=0.50, 105 °C
1992	Asahi	X-1	-	-	-	0.37	540 MHz·10m AN,LED=0.50
1992	Asahi	X-2	-	-	-	0.28	>1.000 MHz·10m AN,LED=0.50
1992	Höchst	EP51	970	190	650	St.	90 MHz·100m With 650 nm LED
1992	Mitsubishi	Eska Premier	1000	135	650	0.51	Hasta 85 °C
1993	CIS	Sveton MN-Series, Grade U	200-600	150	650	0.45	Hasta 70 °C
1993	CIS	Sveton MN-Series, Grade U	200-1000	120	650	0.48	Hasta 70 °C
1994	Sumitomo	n.a.	480	150	650	0.51	200 MHz·50m Δn=0.055
1998	Mitsubishi	n.a.	1000	110	650	0.47	80 MHz·100m
2002	Dig. Optr.	n.a.	1000	-	-	0.5	Anunciado en 2003
2003	Luvantix	SI type	1000	160	650	0.4	Ancho de banda de 200 MHz
2004	Nuvitech	Nuvilight	1000	250	650	0.38	Para iluminación
2005	Luceat	SI-Type	1000	150	650	0.46	30 MHz·100m
2005	Nanoptics	A-POF	1000	100	650	-	Concepto
2005	Huiyuan	SI-POF	1000	300	650	-	Coextrusión

2006	Luceat	SI-POF	1000	135 65	650 520	0.5	Desde preforma
------	--------	--------	------	-----------	------------	-----	----------------

Tabla 1.1 Últimas publicaciones sobre SI-POF

1.3.1. Formatos de Modulación

Una vez planteada la información general sobre los sistemas SI-POF, a continuación se describen los formatos de modulación más empleados en estos sistemas.

Un rango de frecuencias es asignado para cada medio de comunicación. Aquello que se quiere transmitir y no está comprendido en ese rango de frecuencias necesita ser alterado. Esta alteración es la modulación. Es decir, el mensaje se modula, se envía de modulado, y al llegar al destino se demodula para obtener el mensaje original. Los formatos de modulación más empleados hoy en día en las SI-POF son: QAM (*Quadrature Amplitude Modulation*) [9,10], DMT (*Discrete MultiTone*) y PAM (*Pulse Amplitude Modulation*).

Modulación QAM

Este tipo de modulación consiste en transportar dos señales independientes, desfasadas 90° entre sí. La señal modulada es la suma de ambas señales. Al llegar al receptor, ambas señales se demodulan de forma independiente debido a la ortogonalidad de las portadoras. [referencia?]

Modulación CAP

La modulación CAP (por sus siglas en inglés *Carrierless Amplitude Phase Modulation*) es una modulación QAM pero sin portadora. Al no tener potencia DC, esta modulación es perfecta para canales acoplados de AC. La mayoría de los sistemas CAP se aplican en el dominio digital [11]. Mientras que al principio las modulaciones CAP y DMT coexistieron, los organismos encargados de la estandarización se decantaron por la modulación DMT, por lo que hoy en día no son sistemas muy utilizados [12].

Modulación DMT

Este tipo de modulación consiste en la multiplexación de un conjunto de ondas portadoras de diferentes frecuencias y rompe el espectro de frecuencias en múltiples subportadoras, cada una de las cuáles, es ortogonal a las otras. Cada una de estas subportadoras implementa una modulación QAM o PSK manteniendo las velocidades de transmisión similares a las de una sola longitud de onda con el mismo ancho de banda. Esto permite una transmisión y recepción más sencillas.

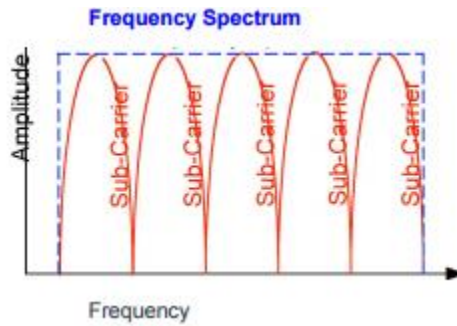


Figura 1.3 Modulación DMT

La ortogonalidad permite una alta eficiencia espectral, casi toda la frecuencia disponible puede ser utilizada. Este tipo de modulación requiere una sincronización de frecuencia muy precisa entre el transmisor y el receptor, ya que con una desviación en esta frecuencia los subcanales dejarían de ser ortogonales, provocando una interferencia.

Modulación PAM

En este caso el tipo de modulación escogido es la modulación de amplitud de pulso (PAM, por sus siglas en inglés *Pulse Amplitude Modulation*). Como su propio nombre indica PAM es la transmisión de datos mediante la variación de la amplitud de los pulsos individuales en una secuencia temporizada regularmente. Dependiendo de las diferentes amplitudes posibles se llama modulación M-PAM. Para que la señal de salida pueda ser digital las amplitudes del pulso tiene que ser una potencia de 2, así que en el caso de 16-PAM hay 2^4 posibles amplitudes de pulsos discretos.

En este tipo de modulación la señal se convierte en pulsos separados por un mismo periodo de tiempo. De esta manera, el demodulador detecta la amplitud de cada periodo y genera de nuevo la señal originalmente transmitida.

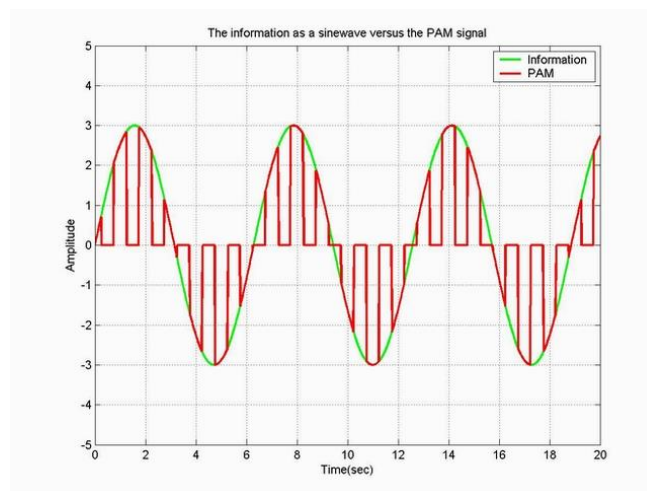


Figura 1.4 Modulación PAM

1.3.2. Evaluación de la tasa de bits en error

En este apartado, se va a describir qué es la tasa de bits en error. También se van a tratar los BERT (Bit Error Rate Tester) así como su empleo hoy en día en los sistemas SI-POF.

La tasa de bits en error, o BER (por sus siglas en inglés *Bit Error Rate*) se trata de una forma para medir el rendimiento de un sistema de comunicación que compara los bits al final de un enlace con los originalmente transmitidos. Su definición es bastante simple.

$$BER = \frac{\text{Errores}}{N^{\circ} \text{ total de bits}}$$

El ruido es de lo que más afecta a una buena transmisión con un BER bajo. Una forma de disminuir la densidad del ruido espectral es disminuir el ancho de banda, pero esta reducción está limitada a la velocidad a la que se quiera transmitir los datos. También se puede aumentar la energía de cada bit, pero esto significaría aumentar la potencia de transmisión, lo que podría ocasionar interferencias con otros sistemas. También se puede aumentar la energía por bit disminuyendo la velocidad de transmisión, pero se pierde capacidad.

BERT (Bit Error Rate Tester)

La medida del BER es bastante sencilla, ya que consiste en mandar una serie de datos conocidos a través de un canal de comunicación y comparar los datos recibidos con los datos enviados. Teóricamente para comprobar el BER de un sistema, sería necesario enviar un número infinito de números y compararlos con los infinitivos números recibidos, ya que los errores ocurren de forma aleatoria. Obviamente, esto no es factible. En este TFG no se llegan a números demasiados altos debido a las limitaciones existentes. Cuanto mayor sea el número de bits transmitidos, mayor fiabilidad tendrá el BER obtenido.

Hoy en día hay varios equipos de instrumentación electrónica que hacen BER. Sin embargo este tipo de equipos comerciales no son adecuados para sistemas SI-POF, ya que actualmente en POF no hay estándares de transmisión y se utilizan anchos de banda que pueden estar muy por debajo de los que utilizan los BERT comerciales. Además muchos de ellos tienen un rango de frecuencias a partir de más de 100 MHz, y los sistemas SI-POF no soportan estas velocidades. También, muchos de ellos no dejan utilizar formatos de modulación experimentales como los que se utilizan en sistemas SI-POF, lo que lo limita bastante. A continuación se puede encontrar una lista con este tipo de equipos de las principales marcas del mercado.

Marca	Modelo	Frecuencia	Modulation
Tektronix	BitAnalyzer	1.5-1.6 GHz	-
Keysight	J-BERT M8020A	256 MHz (mín.)	Triangular and arbitrary
Anritsu	BertWave MP2100B	0.1 GHz (mín.)	DP-QPSK, PAM

Tabla 1.2 Equipos comerciales BERT

Es cierto que existen muchos equipos en el mercado que hacen BERT, pero normalmente los equipos que hacen esta medida, no sólo hacen esa, si no varias diferentes, como es en el caso de las marcas Viavi (JDSU) y Exfo. Eso, obviamente, encarece el valor del equipo, ya que realiza más tests de los que se necesitan para obtener el BER.

1.3. Estado del arte

En el Estado del Arte se van a tratar los últimos sistemas de comunicación de SI-POF tanto de una sola longitud de onda como de varias. Se van a comparar los mismos para tener una mejor visión de los últimos avances de estos sistemas.

Sistemas de transmisión SI-POF con una sola longitud de onda

Generalmente la transmisión a través de SI-POF se hace con una longitud de onda de 650 nm, alcanzando velocidades de 100 Mb/s en enlaces de 275 m [13,14]. Existen en el mercado sistemas basados en enlaces SI-POF que son capaces de proporcionar velocidades de 1Gb/s en 50 m. El problema de utilizar un sistema con una sola longitud de onda es la limitación en el ancho de banda. Para aumentar el ancho de banda la única posibilidad es aumentar la velocidad de transmisión de los datos, lo que disminuye la relación señal-ruido y, por lo tanto, las mejoras que se pueden conseguir gracias a esto son limitadas.

En la siguiente tabla se pueden observar una comparativa entre los últimos sistemas creados.

Fecha	Velocidad (Gb/s)	Longitud (m)	Longitud de onda (nm)	Modulación	Bidireccional	BER
06/11 [15]	1	75	-	-	Sí	-
03/13	3	25	-	PAM-8	-	-
09/14 [16]	5	25	650	PAM-32	Sí	$<10^{-12}$
12/14 [17]	5.56	1	475	PAM-4	-	-
02/15 [18]	8	10	450	PAM-16	Sí	-
08/15 [19]	6.25	10	450	PAM-16	-	$<10^{-3}$
19/15 [20]	10	10	450	PAM-32	Sí	$<10^{-3}$

Tabla 1.3 Sistemas con una longitud de onda

Sistemas de transmisión SI-POF multiplexados en longitud de onda

Una opción para aumentar los ratios de datos de los enlaces basados en la SI-POF es usar varias portadoras en una sola POF para transmitir de forma paralela varios canales de comunicación [21]. Como los canales están bastante alejados en comparación a las comunicaciones ópticas de larga distancia, se trata de una *Multiplexación por división de onda* (WDM por sus siglas en inglés *Wavelength Division Multiplexing*). Cualquier sistema basado en arquitecturas WDM requiere, como mínimo, un multiplexor, que combina las diferentes longitudes de onda en una sola fibra, y un demultiplexor al final del enlace, para volver a separar las diferentes señales, que añaden pérdidas de inserción al sistema. Debido al material plástico del que está compuesto este tipo de fibra, sólo se puede trabajar en el rango de frecuencias visible [1] se utiliza el término WDM visible. Las actuales propuestas de WDM visible sobre SI-POF están basadas en redes espectrales con canales entre 400 nm y 700nm empleando como emisor LEDS o láseres [22].

En la siguiente tabla se pueden observar una comparativa entre los últimos sistemas creados.

Fecha	Velocidad (Gb/s)	Longitud (m)	Número de canales	Longitud de onda (nm)	Modulación	BER
08/12 [23]	10.7	50	3	405, 515, 615	-	$<10^{-3}$
03/14 [24]	21.4	50	6	-	DMT	$<10^{-3}$
05/14 [25]	8.26	100	4	405, 450, 515, 639	DMT	$<10^{-3}$
05/14 [25]	14.77	50	4	405, 450, 515, 639	DMT	$<10^{-3}$
12/15 [26]	11	10	3	420, 450, 515	PAM	$<10^{-3}$
03/16 [27]	10.7	100	6	405, 442, 459, 490, 515, 660	DMT	$<10^{-3}$

Tabla 1.4 Sistemas con varias longitudes de onda

1.4. Objetivos específicos y alcance del proyecto

El objetivo final es la realización de una guía práctica para la evaluación del BER en sistemas M-PAM sobre SI-POF. Es decir, sin necesidad de material adicional, se pueden evaluar estos parámetros gracias a este TFG.

De esta manera, se implementa un sistema en el que un generador de ondas arbitrario envía una señal digital modulada por amplitud de pulsos (M-PAM) y codificada. Esta señal se envía a través de un enlace de fibra óptica de plástico de salto de índice (SI-POF), para su posterior detección y análisis de calidad, en términos de la tasa de error de bit (BER).

La generación, modulación, demodulación, y análisis de la señal digital se realizará con la ayuda de programas como MATLAB y SIMULINK; mientras que la transmisión de la señal eléctrica y la detección de la señal óptica se realizarán con la ayuda de un generador de onda arbitraria (de hasta 500 MHz) y un osciloscopio (en combinación de un foto-detector de alta velocidad), respectivamente.

Después del desarrollo e implantación del sistema, se desarrollarán varios casos de estudios para comprobar la efectividad del sistema previamente desarrollado.

Finalmente todo el sistema ha sido implantado en un cable coaxial, no ha llegado a probarse en un enlace de SI-POF por falta de tiempo, hubiera sido el siguiente paso.

Capítulo II: Planificación

Para alcanzar los objetivos propuestos en este Trabajo de Fin de Grado se ha diseñado y seguido el plan de trabajo que se detalla a continuación.

Estudio del Estado del Arte

Fue la primera etapa del proyecto, en la que se adquirieron los conocimientos requeridos para llevar a cabo este proyecto. En este caso concreto, dicha formación trató sobre modulación digital y sobre la estimación del BER mediante Matlab a través de un canal AWGN (*Additive White Gaussian Noise*). Se realizaron diferentes simulaciones con este software. Además, la formación también estuvo enfocada al uso de los diferentes equipos implicados en el proyecto.

Diseño y desarrollo del sistema en Simulink

Se llevó a cabo toda la simulación en Simulink y se ajustaron todos los parámetros hasta que se consiguió transmitir los datos. En esta etapa, no se utilizó ningún hardware.

Experimentación con tarjetas de National Instruments

Después de conseguir transmitir los datos en Simulink se llevó la simulación al laboratorio. Se sustituyó el canal AWGN por unas tarjetas de National Instruments, aunque el resultado no fue el esperado, por lo que se decidió cambiar a otro sistema de comunicaciones.

Diseño y desarrollo del sistema con el generador de onda arbitraria (sistema final)

Se decidió cambiar las tarjetas de NI debido a la limitación que imponían (48KS/s). Por ese motivo, se diseñó un nuevo sistema en el los datos se generaban en Matlab, pero se transmitían desde un generador de onda arbitraria y se recuperaban los datos gracias a un osciloscopio.

Experimentación con el sistema final

Una vez desarrollado el sistema final, se llevaron a cabo varios experimentos con diferentes datos para compararlos entre ellos y destacar el caso en el que había mejor transmisión.

Documentación y escritura de la memoria

Por último, se ha realizado la memoria del proyecto, para documentar el mismo y registrar toda la información necesaria para hacerlo comprensible y poder desarrollarlo sin necesidad de ningún otro material.

2.1. Diagrama de Gantt

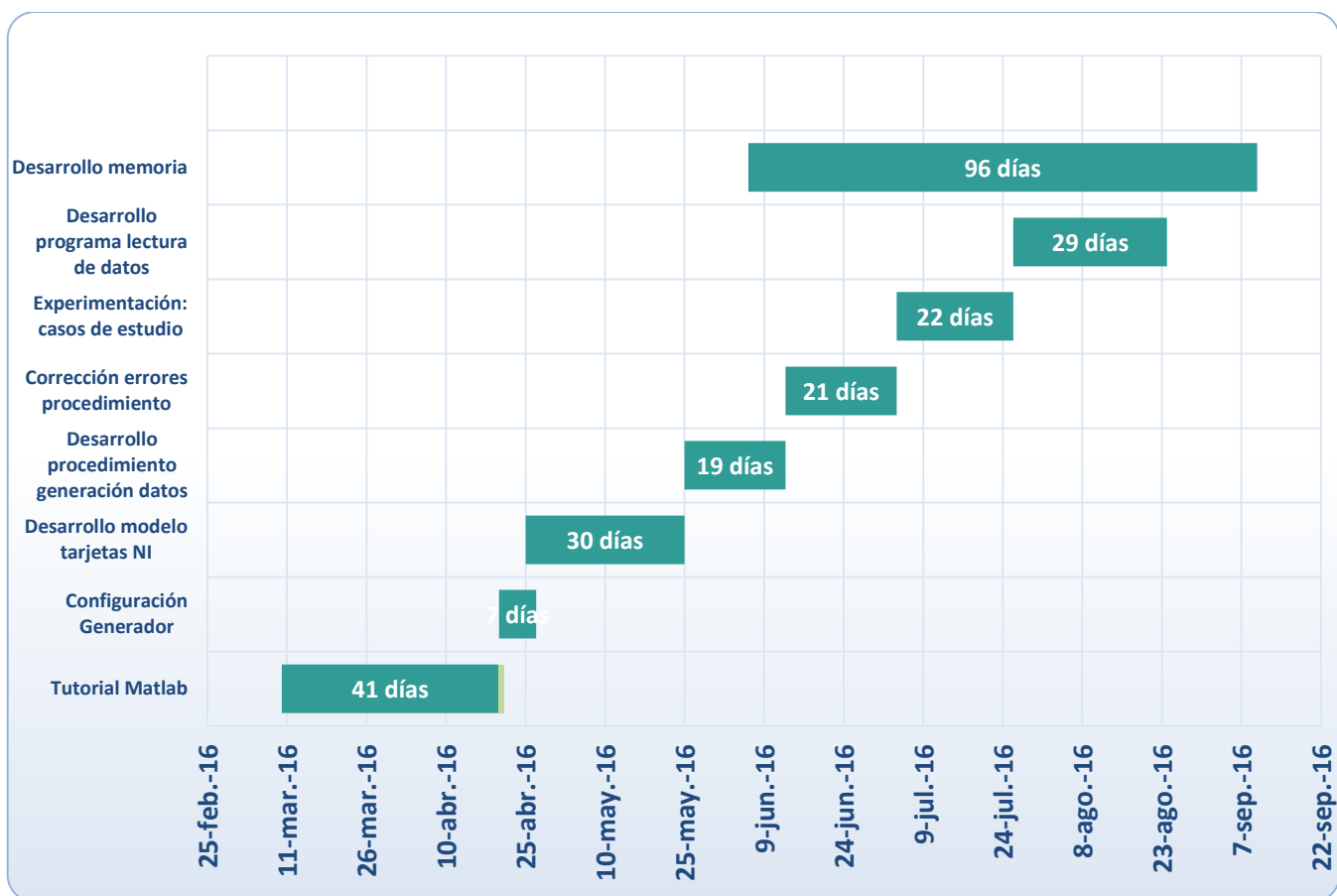


Figura 2.1 Diagrama de Gantt

Capítulo III: Simulaciones y procedimientos

En este capítulo se va a desglosar el desarrollo, primero, y la procedimiento, después, de los diferentes sistemas propuestos. Al principio se llevan a cabo simulaciones para luego aplicarlas al sistema final.

3.1. Diagrama de bloques general

Esta primera simulación trata sobre un sistema basado en Simulink en el que la transmisión se hace mediante las tarjetas de National Instruments descritas más abajo.

Generación de la señal de transmisión

En este apartado, se explica cómo se genera la señal que se va a transmitir mediante las tarjetas.

- *Generalidades sobre modulación y codificación*

Para simular PAM en Matlab se realizó el siguiente ejemplo para modular y demodular números enteros arbitrarios y verificar que los datos recibidos eran igual a los enviados. Se trata de un ejemplo muy básico para definir y entender los conceptos básicos así como el procedimiento.

```
M = 12;  
dataIn = randi ([0 M-1, 100, 1]);
```

donde M es orden de modulación y $dataIn$ son 100 símbolos de datos M-ary generados aleatoriamente. La idea de los símbolos es juntar 2 o más bits en un símbolo y transmitir los símbolos. Hay que tener en cuenta que los símbolos tienen que ser una potencia de 2 bits:

$$m = 2^k$$

donde m es el número de símbolos y k el número de bits. El siguiente paso es modular y demodular los datos.

```
modData = pammod (dataIn, M);  
dataOut = pamdmod (modData, M);
```

donde $modData$ son los datos arbitrarios modulados y $dataOut$ es la salida del demodulador, con los datos originales recuperados. Para comprobar que no ha habido errores de transmisión y que efectivamente los datos a la salida del demodulador son los mismos que los enviados, se comparan los 5 primeros datos enviados con los 5 recibidos.

```
[dataIn(1:5) dataOut(1:5)]
ans=
     9     9
    10    10
     1     1
    10    10
     7     7
```

Como se puede ver, los 5 primeros datos enviados son los mismos que los recibidos. Ahora se verifica que no hay ningún error en la secuencia de los 100 números.

```
symErrors = symerr(dataIn, dataOut)
symErrors=
0
```

Para minimizar los futuros posibles errores a la hora de implementar la modulación con un circuito se implementa el código Reed-Solomon. Se trata de un código detector y corrector de errores de transmisión. Estos errores se deben, generalmente, a ruido. Con este propósito se simula una transmisión Reed-Solomon usando un 8-PAM sobre un canal AWGN (*Additive White Gaussian Noise*). El AWGN es un modelo de ruido básico para imitar el efecto de diversos procesos aleatorios que se producen en la naturaleza. Para ello, definimos los parámetros de la simulación.

```
M      = 8;
bps     = log2(M);
N       = 7;
K       = 5;
```

donde M es el orden de modulación, bps es la cantidad de bits por símbolo, N es la longitud de la palabra de código y K es la longitud del mensaje. El siguiente paso es crear el modulador, demodulador, el canal AWGN y la tasa de error.

```
mod      = comm.PAMModulator('ModulationOrder',M, 'BitInput',true);
demod    = comm.PAMDemodulator('ModulationOrder',M, 'BitOutput',true);
chan     = comm.AWGNChannel('BitsPerSymbol',bps);
err      = comm.ErrorRate;
```

Ahora hay que crear el codificador y decodificador Reed-Solomon que acepte entradas de bits.

```
enc = comm.RSEncoder('BitInput',true, 'CodewordLength',N,
'MessageLength',K);
dec = comm.RSDecoder('BitInput',true, 'CodewordLength',N,
'MessageLength',K);
```

Se ajusta el intervalo de E_b/N_0 y se inicializa la matriz de estadísticas de error. E_b/N_0 es la densidad de energía en el ruido espectral y se mide en dB, ya que E_b es la cantidad de energía por bit y N_0 es el ruido de densidad espectral [Watts/Hz].

```
ebnoVec = (3:0.5:8)';
errorStats = zeros(length(ebnoVec),3);
```

Por último, se hace la estimación del BER (Bit Error Rate) para cada valor de E_b/N_0 . Se simula hasta que se encuentre 100 errores o se reciban 10^7 bits. La simulación incluye codificación, modulación, demodulación y decodificación.

```
for i = 1:length(ebnoVec)
    chan.EbNo = ebnoVec(i);
    reset(err)
    while errorStats(i,2) < 100 && errorStats(i,3) < 1e7
        data = randi([0 1], 1500, 1);           %Generación de datos
        encData = step(enc, data);               %Codificación RS
        modData = step(mod, encData);            %Modulación
        rxSig = setp(chan, modData);             %Señal mediante AWGN
        rxData = step(demod, rxSig);             %Demodulación
        decData = step(dec, rxData);             %Decodificación RS
        errorStats(i,:) = step(err, data, decData); %Colecta de
    errores
end
end
```

Para terminar es necesario ajustar una curva a los datos del BER y dibujarla.

```
berCurveFit = berfit(ebnoVec, errorStats(:,1));
berNoCoding = berawgn(ebnoVec, 'pam', 8, 'nondiff');
semilogy(ebnoVec, errorStats(:,1), 'b*', ...
    ebnoVec, berCurveFit, 'c-', ebnoVec, berNoCoding, 'r')
ylabel('BER')
xlabel('Eb/No (dB)')
legend('Data ', 'Curve Fit', 'No Coding')
grid
```

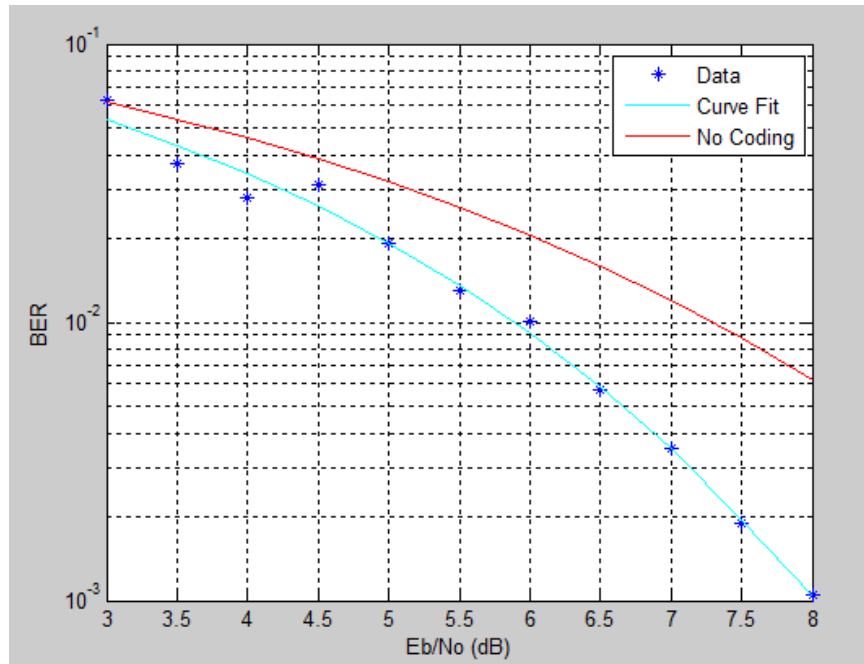



Figura 3.1 BER vs E_b/N_0 (dB)

Como se puede ver en la Figura, usando la codificación Reed-Solomon se consigue mejorar el E_b/N_0 requerido para conseguir 10^{-2} de BER aproximadamente 1.4 dB.

Implementación en Simulink

Una vez simulado el sistema, se experimenta en el laboratorio.

- **Pruebas con la tarjeta NI 6009**

Inicialmente se diseñó el siguiente sistema de transmisión, en el que el canal AWGN se sustituye por las tarjetas de National Instruments 6009.

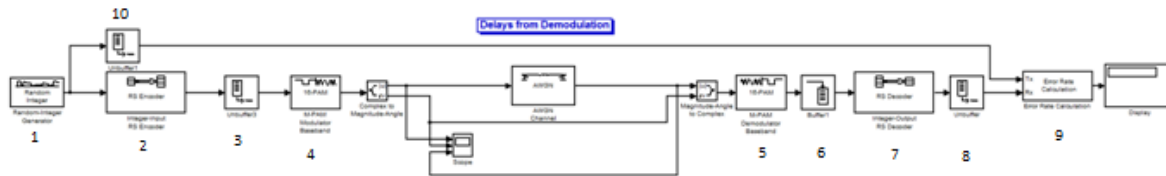


Figura 3.2 Esquema sistema de transmisión con tarjetas NI

El Random-Integer Generator¹ genera enteros arbitrariamente uniformemente distribuidos en el rango $[0, M-1]$. En este caso se le ha asignado a M un valor de 3 y la salida consiste en tramas de 5 muestras. La primera salida de este generador va conectada a la entrada del codificador. La segunda salida va a un Unbuffer¹⁰ para convertir los vectores de salida en escalares independientes y de ahí va al Error Rate Calculation⁹.

El RS-Encoder² codifica el vector de entrada usando un (7,5) Reed-Solomon, por lo que N (longitud de la palabra código) es 7 y K (longitud del mensaje es 5). Cada grupo de 5 elementos representa una palabra a codificar y cada símbolo tiene que tener $\log_2(N+1)$ bits. A la salida del codificador va conectado un Unbuffer³.

El Unbuffer³ convierte los vectores de entrada en salidas independientes. El Unbuffer³ no produce ningún retraso, ya que el tiempo de entrada es igual al tiempo de salida. La salida del Unbuffer³ va conectado a la entrada del 16-PAM Modulator.

El 16-PAM Modulator⁴ modula los datos recibidos del Unbuffer³. A la salida del modulador se separa la parte real de la imaginaria. Esta parte real es la que se envía a través de la tarjeta de National Instruments.

Después de pasar los datos por la Tarjeta y visualizarlos mediante un osciloscopio, se le añade la misma fase que se le retiró a la salida del modulador para introducir los datos recibidos de la tarjeta en el 16-PAM Demodulator⁵, el cual demodula estos datos para llevarlo a un Buffer⁶.

El Buffer⁶ convierte los escalares del demodulador en vectores de 7 muestras. El uso del Buffer⁶ produce un pequeño retraso, el cual luego se tiene en cuenta a la hora de calcular el BER.

La salida del Buffer va conectada al RS Decoder⁷, el cual emplea una decodificación (7,5) Reed-Solomon. Este decodificador acepta a la entrada vectores de 7 elementos. Cada grupo de 7 elementos es una palabra a decodificar. Igual que en codificador, cada símbolo tiene que tener $\log_2(N+1)$ bits. De nuevo, la salida de decodificador va conectada a la entrada de un Unbuffer⁸ para convertir los vectores en escalares independientes.

La salida del Unbuffer⁸ va al Error Rate Calculation⁹, donde se compara con la salida del Unbuffer¹⁰ que va conectado al generador de enteros. La salida de este bloque es un vector de tres elementos con el ratio de errores, seguido de los errores detectados y de todos los símbolos comparados. En este bloque hay que especificar el retraso producido por el Buffer⁶.

Asignando a M un valor de 3 y sabiendo que en el codificador M tiene que ser $\log_2(N+1)$, se le asigna un valor de 7 a N. Como la salida del generador arbitrario de enteros está basada en muestras de 5, el valor de K en el codificador tiene que ser 5.

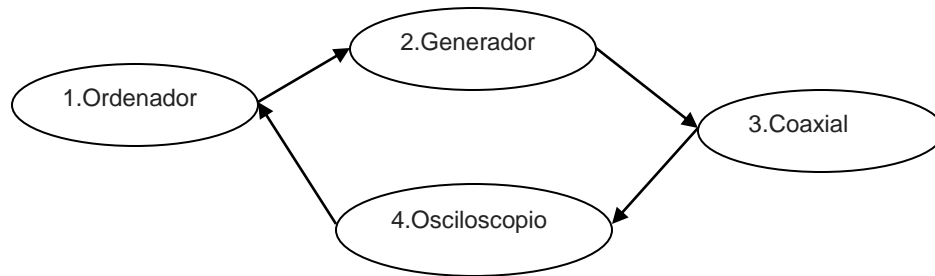
Después de hacer estos cambios seguía habiendo una cantidad de errores cercana al 68 %. Esta cantidad se debe a que al utilizar el codificador, la velocidad de transmisión cambia, ya que se pasa de 5 muestras a 7. Para corregir esos errores, se le asigna al generador arbitrario de enteros una velocidad de muestreo de $7 \cdot 10^{-3}$ s y a la tarjeta de National Instruments de $5 \cdot 10^{-3}$ s. De esta manera, se consigue que no se transmita ningún error, aunque es cierto que la frecuencia es muy baja.

Después de ajustar los tiempos de muestreo de generador aleatorio de enteros y de la tarjeta de National Instruments, el ancho del bit es de 9.2 ms. El próximo paso es sustituir la tarjeta por otra de más potencia para conseguir que el ancho del bit llegue a los 9 ns (ya que el objetivo de este trabajo es probar sistemas SI-POF con anchos de banda de aproximadamente 100 MHz). Debido a la limitación en frecuencia que supone el uso las tarjetas NI USB 6009 se debe cambiar a otro sistema, ya que se está buscando una frecuencia en la que la SI-POF pueda trabajar.

3.2. Sistema de transmisión y detección de señales M-PAM propuesto

Tras comprobar que las tarjetas disponibles en el laboratorio no eran capaces de alcanzar las velocidades deseadas, es necesario realizar otro procedimiento. Además, el trabajo realizado con SIMULINK fue una herramienta para entender el sistema de transmisión y los conceptos básicos de señales M-PAM en un enlace SI-POF. Por eso, en este apartado se procede con la explicación del procedimiento para la obtención de los datos recibidos y su comparación con los datos originalmente enviados. De esta manera se pueden observar el número de errores generados debido a la transmisión de los datos.

Esquema



En primer lugar, es necesario generar los datos. Este proceso, así como su modulación y codificación se hace en un ordenador en MATLAB. Dicho ordenador está conectado mediante un cable USB al Generador de funciones arbitrarias. El generador sólo admite datos en un archivo .txt por lo que es necesario crear este tipo de archivo con los datos a transferir (uno por fila).

Una vez que tenemos los datos en el archivo .txt, mediante el software del generador, seleccionamos el archivo como tipo de onda. El generador, a su vez, está conectado mediante 1 metro de cable coaxial al osciloscopio, por lo que cuando se hace click en el botón de “play” se mandan los datos.

Una vez que el osciloscopio está recibiendo los datos, es necesario hacer una captura de pantalla de manera que se capturen todos los datos enviados. Para hacerlo, simplemente basta con fijarse en la cadena de ceros que hay al principio de la señal. Una vez que guardamos la captura en el disquete, es necesario abrir el archivo Excel en el que se guardan en el ordenador y dividir los datos en dos columnas, ya que el osciloscopio guarda en una sola columna el tiempo y el valor del bit.

Ya con los datos en un archivo Excel, se leen, decodifican y demodulan mediante MATLAB. Estos datos se comparan con los generados en primer lugar para hacer el análisis de calidad de la señal en términos de BER.



Figura 3.3 Sistema completo

Descripción detallada de componentes

A continuación se detallan los componentes empleados en este procedimiento:

- Ordenador: con Windows XP para poder ejecutar el software del generador de ondas y con disquetera para poder leer los datos del osciloscopio,

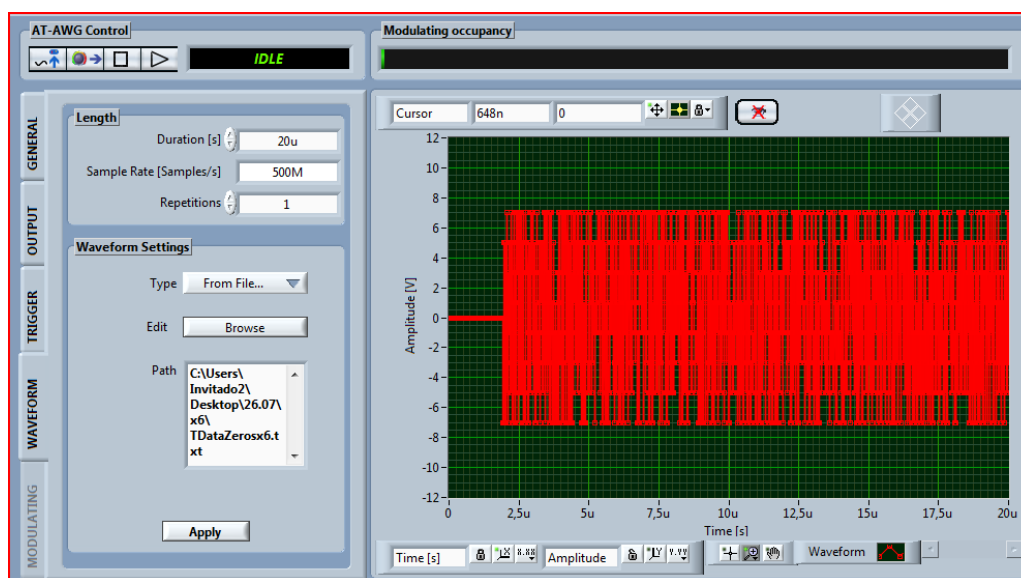


Figura 3.4 Software generador de ondas

- Generador de señal: AT-AWG500. Se trata de un generador de ondas arbitrarias de 500 MHz.



Figura 3.5 Generador de ondas

- Cable coaxial: 1 metro
- Osciloscopio: Tektronix TDS3052. Osciloscopio de 500 MHz y 2 canales.



Figura 3.6 Osciloscopio

Procedimiento de configuración

En este apartado se explica el procedimiento completo, desde la generación de los datos hasta el cálculo del BER.

- *Generación de datos*

Para generar los datos en cada caso de estudio, es necesario, primero, definir el sistema de transmisión. En este caso, hay que definir, como ya se hizo con anterioridad, los parámetros.

```
M = 8;  
bps = log2(M);  
N = 7;  
K = 5;
```

Se crean el modulador y el demodulador.

```
mod = comm.PAMModulator('ModulationOrder',M,'BitInput',true);  
enc = comm.RSEncoder('BitInput',true,'CodewordLength',N,'MessageLength',K);
```

Por último, se generan, modulan y codifican los datos que se van a transmitir al generador.

```
data = randi([0 1],d,1);  
encData = step(enc,data);  
modData = step(mod,encData);
```

donde d es el número de datos a generar. Para calcular este número, es necesario tener en cuenta la relación que guardan $data$ y $modData$.

$$modData = data * \frac{7}{5} * \frac{1}{3}$$

Para poder obtener unos buenos datos recibidos es necesario hacer un sobremuestreo. De esta manera cuando se capturen los datos del osciloscopio se puede reconstruir la señal perfectamente. El empleo del sobremuestreo mejora la resolución y reduce el ruido y la distorsión.

Dependiendo del sobremuestreo que se quiera hacer, el número de datos a generar es diferente, como se verá más adelante. Es importante calcular bien $data$ y $modData$ para que sean números enteros. Para hacer este sobremuestreo una vez generados los datos aleatorios, es necesario el siguiente código.

```
for i = 1 : modData  
    TData((i-1)*n+1:i*n)=modData(i)  
end;  
TData = TData'
```

De esta manera, se crea *TData*, que son los datos modulados una vez hecho el sobremuestreo. A la hora de utilizar el código, *modData* es necesario sustituirlo manualmente por el número de datos que contiene y *n* por el número de bits que se quiere que contenga cada dato. Es decir, si se quiere implementar un sobremuestro x3, es decir, repita durante 3 periodos de muestreo del generador de funciones, *n* debe ser sustituido por 3. Entonces, el sobremuestreo permite controlar la velocidad de transmisión. La última línea del código simplemente transforma el vector de datos de horizontal a vertical.

El último paso antes de transferir los datos al generador es añadir los ceros pertinentes. Es necesario añadir ceros al principio o al final de la señal enviada para poder identificar cuando comienza. Saber esto posibilita la posterior comparación con la señal enviada. Sin embargo, después de realizar algunas pruebas se determinó que no es la mejor opción, ya que la secuencia de datos de la señal enviada podría comenzar con varios ceros seguidos.

Hay que tener en cuenta que el osciloscopio utilizado en este TFG sólo permite recuperar 10.000 datos, por lo que los datos sobremuestreados más los ceros deben, idealmente, ser 10.000.

```
TDataZeros = [zeros(z,1); TData]
```

donde *z* es el número de ceros que se deben añadir a *TData* para llegar los 10.000 datos.

- *Transferencia de datos al generador*

Una vez que se dispone de *TDataZeros* es necesario transferir los datos al generador. El generador empleado permite la lectura de los datos desde un archivo *.txt*. Por ese motivo es necesario exportar los *TDataZeros* en un archivo de texto.

Una vez que este archivo está creado, se genera una señal con estos datos. En este caso al seleccionar 500MSamples/s los máximos puntos que va a transmitir son 10.000, por lo que el programa no haría ningún sobremuestreo adicional. Para que la función que se genera sea correcta hay que seleccionar *Coerce* la opción en *Interpolation*.

- *Captura de datos*

Una vez que se generan los datos, se transmiten mediante un cable coaxial de 1 metro y se capturan con el osciloscopio. Para capturar los datos de Rx es necesario hacer una captura de pantalla de manera que se vea un periodo completo de la señal enviada. De esta manera, se pueden guardar todos los datos de la señal en un archivo de Excel. Este procedimiento podría ser automatizado programando e interconectando el trigger del generador de funciones con el del osciloscopio. El osciloscopio permite guardar la captura seleccionada en un disquete. Los datos guardados son ambos ejes, es decir, tanto el tiempo como el valor de la señal en ese momento.

- *Cálculo del número de bits en error*

Se han realizado cuatro casos de estudios diferentes, cada uno con un número diferente de datos debido a que cada uno tiene un número diferente de puntos por bit.

Los datos de los cuatro experimentos se encuentran en un Excel llamado "Total.xlsx" con cuatro pestañas: "x3" para el caso con 3 puntos por bit, "x4" para el de 4 puntos por bit, "x5" para el de 5 puntos por bit y "x6" para el de 6 puntos por bit.

Lectura de datos

La primera acción necesaria es leer tanto los datos transmitidos como los generados, colocados en las columnas F y H respectivamente.

```
file = n
Tx0   = xlsread('Total.xlsx', file, 'F6:F10005');
Rx0   = xlsread('Total.xlsx', file, 'H6:H10005');
```

donde *file* es el número de puntos por bit (para acceder a la pestaña correcta del Excel) y 10005 es la última fila del Excel en la que se puede encontrar un dato. Para ejecutar el programa de forma correcta, es necesario cambiar la *n* de manera manual para cada caso de estudio ('x4', 'x5' o 'x6'). Para que el programa de Matlab pueda leer los datos del archivo Excel, este archivo debe estar en la misma carpeta en la que se encuentra el programa ejecutado.

A continuación se procede con la creación de las gráficas para ambas señales (ver Anexo II).

Al comparar las dos gráficas de manera conjunta se puede observar si hay o no desfase entre ambas señales y, en caso de haberlo, cuantificarlo. Una vez que se dispone de estas gráficas, se puede cuantificar la frecuencia de muestro. Para cuantificar esta frecuencia de Tx basta con analizar la columna de tiempo obtenida de la lectura del osciloscopio:

$$FsRx0 = \frac{1}{t2 - t1}$$

Una vez obtenida esta frecuencia se puede observar en la *Gráfica con datos de transmisión y señal recibida* la cantidad de puntos y periodos de Rx y compararla con los puntos y periodos de Tx para determinar su frecuencia de muestreo.

```
FsRx0   = 500e6;
FsTx0   = 500e6;
```

Búsqueda de la muestra inicial

El siguiente paso es el de encontrar la muestra inicial para comparar ambas señales. La idea de colocar ceros al inicio de la señal de transmisión es la de facilitar la localización de la muestra inicial. Sin embargo, después de realizar algunas pruebas se determinó que no es la mejor opción, ya que,

al generar los datos de forma aleatoria, la señal podría empezar con varios bits de ceros. Como se puede observar en los casos de estudio concretos de este TFG no se produce esta situación.

Para el futuro, se podría implementar un patrón para identificar la muestra inicial. En este caso se a utilizar la señal Tx como patrón. Para ellos, se procede con la normalización de las señales Tx y Rx.

```
TxC = Tx0/max(abs(Tx0));  
RxC = Rx0/max(abs(Rx0));
```

A continuación se hace una correlación rotando la señal de recepción (Rx) y comparándola con la de transmisión (Tx).

```
C = zeros(size(TxC));  
  
for i = 1 : length(TxC)  
    C(i) = corr(TxC,RxC([i+1:end, 1:i]));  
end
```

Una vez hecho esto, se procede a encontrar el punto en el que ambas señales están superpuestas.

```
i = find(C==max(C));  
RxC = RxC([i+1:end, 1:i]);
```

Y se crea una gráfica con ambas señales superpuestas (ver Anexo II).

Eliminación de ceros adicionales

En este paso se procede con la eliminación de los ceros adicionales. Debido a que en cada caso de estudio se añadió un número de ceros diferentes, se crea un programa enfocados los 3 casos de estudio que se tratan en este TFG.

```
if strcmp(file,'x6')  
    num_of_zeros = 970;  
elseif strcmp(file,'x5')  
    num_of_zeros = 480;  
elseif strcmp(file,'x4')  
    num_of_zeros = 368;  
elseif strcmp(file,'x3')  
    num_of_zeros = 235;  
else  
    break  
end  
  
DataTx = TxC(num_of_zeros+1:end);  
DataRx = RxC(num_of_zeros+1:end);  
timeTx = (0:length(DataTx)-1)/FsTx0;  
timeRx = (0:length(DataRx)-1)/FsRx0;
```

se puede observar que la gráfica *Eliminación de ceros adicionales* la señales comienzan en el punto siguiente al que acaban los ceros (ver Anexo II).

Submuestreo

Debido a que después de generar los datos de forma aleatoria se hizo un sobremuestreo para la correcta visualización de los bits en el osciloscopio, ahora hay que hacer el submuestreo, puesto que cada bit está compuesto de varios puntos. De esta manera, se reducen los bits a un único valor. En este paso se genera un programa enfocado a los tres casos de estudio concretos de este TFG. El submuestreo se hace en ambas señales.

```
if strcmp(file,'x6')
    p = 6;
elseif strcmp(file,'x5')
    p = 5;
elseif strcmp(file,'x4')
    p = 4;
elseif strcmp(file,'x3')
    p = 3;
else
    break
end

for i = 1 : length(DataTx)/p
    positions = [(i-1)*p+1, i*p];
    DTx(i)     = mean(DataTx(positions(1):positions(end)));
    tTx(i)     = mean(timeTx(positions(1):positions(end)));
end
```

En el caso de la señal transmitida es más sencillo dado que cada bit está exactamente formado por n puntos idénticos. En el caso de la señal recibida, se puede hacer de dos maneras diferentes. La primera opción es una media ponderada descartando los puntos extremos y cogiendo los datos centrales de cada bit. La segunda opción es hacer una interpolación. Dependiendo de la manera de hacer el submuestreo se obtiene un número de errores diferentes, como se puede observar en los casos de estudio.

```

% Opción 1: media ponderada

for i = 1 : length(DataRx)/p      positions = [(i-1)*p+1, i*p];
    if strcmp(file,'x6')
        DRx(i) = (DataRx(positions(1)+1) + 2*DataRx(positions(1)+2) +
2*DataRx(positions(2)-2) + DataRx(positions(2)-1))/6;
    elseif strcmp(file,'x5')
        DRx(i) = (DataRx(positions(1)+1) + 2*DataRx(positions(1)+2) +
DataRx(positions(2)-1))/4;
    elseif strcmp(file,'x4')
        DRx(i) = (DataRx(positions(1)+1) + 2*DataRx(positions(2)-1)
)/3;
    else
        break
    end

    % Tiempos de muestreo
    tRx(i) = mean(timeRx(positions(1)+1:positions(2)-1));
end

```

```

% Opción 2: Interpolación

for i = 1 : length(DataRx)/p
    positions = [(i-1)*p+1, i*p];
    DRx(i) =
interp1(positions(1):positions(2),DataRx(positions(1):positions(2)),mean(po
sitions));

    % Tiempos de muestreo
    tRx(i) = mean(timeRx(positions(1)+1:positions(2)-1));
end

```

A continuación se procede con la creación de la gráfica para comparar ambas señales una vez hecho el submuestreo (ver Anexo II).

Demodulación y decodificación

Para comparar la señal inicialmente generada con la señal recibida se debe proceder con la demodulación y decodificación de ésta. En el caso de la señal transmitida, no es necesario realizarlo puesto que los datos son conocidos al haber sido creados al principio, por lo que se puede demodular y decodificar o leer directamente del archivo de Excel en el que se encuentran los datos.

```

M = 8;          % Orden de modulación
bps = log2(M);  % Bits por símbolo
N = 7;          % RS Longitud de la palabra código
K = 5;          % RS Longitud del mensaje

```

A continuación se crean el demodulador, el ratio de errores y el decodificador.

```
demod = comm.PAMDemodulator('ModulationOrder',M,'BitOutput',true);  
dec = comm.RSDecoder('BitInput',true,'CodewordLength',N,'MessageLength',K);  
err = comm.ErrorRate;
```

Y se procede con las señales transmitida y recibida

```
demod_DTx = step(demod,(DTx(1:end)+1e-20j));  
data = step(dec,demod_DTx);  
  
demod_DRx = step(demod,(DRx(1:end)+1e-20j));  
decData = step(dec,demod_DRx);
```

Por último, se calculan los errores de transmisión que se han producido.

```
reset(err);  
errorStats(1,:) = step(err,data,decData);  
fprintf('\n')  
display(['Results : BER = ', num2str(errorStats(1)), ' | Errors = ',  
num2str(errorStats(2)), ' | Samples = ', num2str(errorStats(3))])  
fprintf('\n')
```

Lo siguiente es ejecutar este procedimiento para los cuatro casos de estudios.

Capítulo IV: Resultados experimentales

En este capítulo se van a exponer los resultados obtenidos de los casos de estudios desarrollados a partir del sistema escogido en el capítulo anterior.

4.1. Caso de estudio 1: Portadora a 100 MHz (x6)

En el primer caso de estudio se un sobremuestreo x6, es decir, que cada bit está compuesto por 6 puntos. Para producir estos datos, se decide que *data* contenga 3.225 datos. De manera que:

$$modData = 3225 * \frac{7}{5} * \frac{1}{3} = 1505$$

Por lo tanto, a la hora de generar los datos, es necesario especificarlo.

```
data = randi([0 1],3225,1);
```

Se procede con el sobremuestreo x6

```
for i = 1 : 1505
    TData((i-1)*6+1:i*6)=modData(i)
end;
TData = TData'
```

En este caso, *TData* contiene 9.030 datos, por lo que se le añaden 970 más para transmitir 10.000 datos.

```
TDataZeros = [zeros(970,1); TData]
```

A continuación se transfieren los datos al generador y se pasan a través del osciloscopio. Para leer los datos correspondientes a este caso, es necesario seleccionar la pestaña correspondiente del archivo.

```
file = 'x6'
```

Una vez seleccionado el caso, se obtiene la *Gráfica con datos de transmisión y señal recibida*.

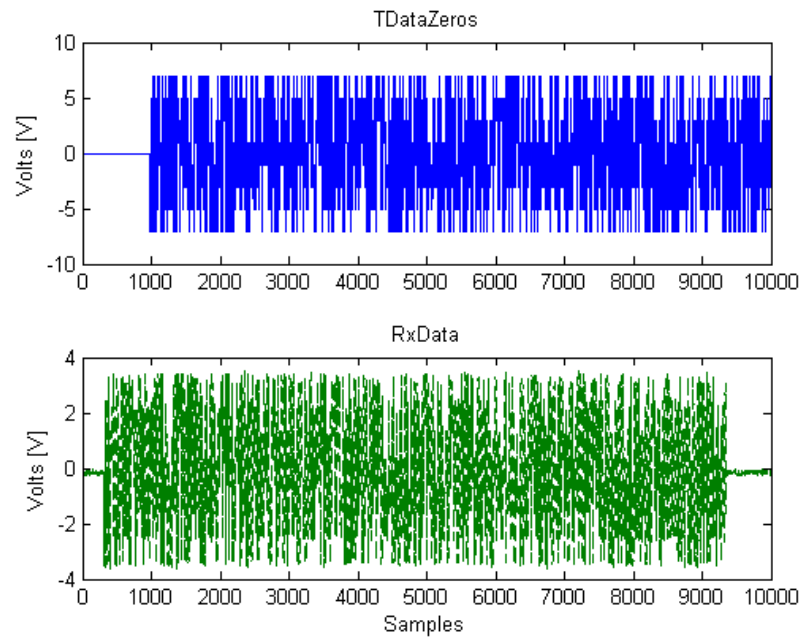


Figura 4.1 Datos de transmisión y señal recibida x6. La Línea sólida corresponde a Tx y la punteada a Rx.

Analizando esta gráfica se observa que ambas señales tiene el mismo número de puntos y periodos, por lo que la frecuencia debe ser la misma. Así mismo, se observa un desfase entre ambas señales, así que se procede a buscar el grupo de ceros para superponerlas.

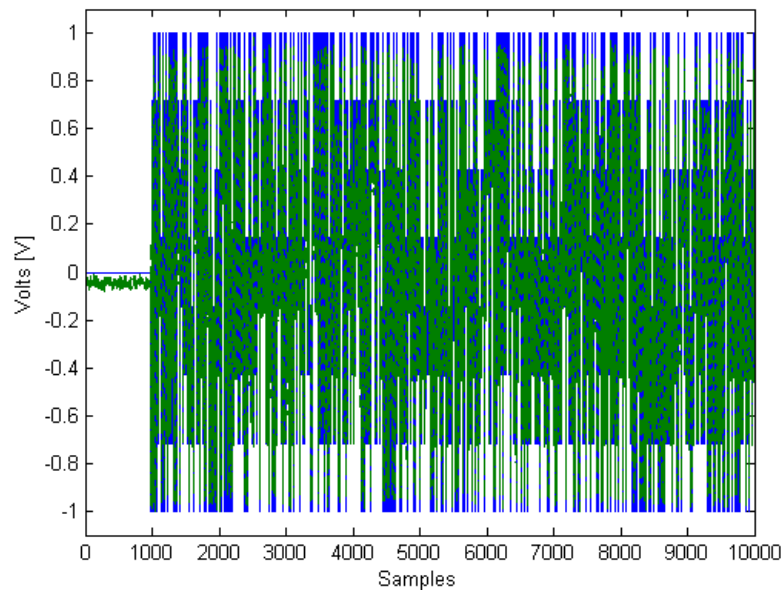


Figura 4.2 Desplazamiento de la señal recibida: correlación x6- La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Desplazamiento de la señal recibida: correlación* se observa que hay un número de ceros antes de la que los datos de las señales. El número de ceros es conocido y, en ese caso, es 970. A continuación se procede con la eliminación de los mismos.

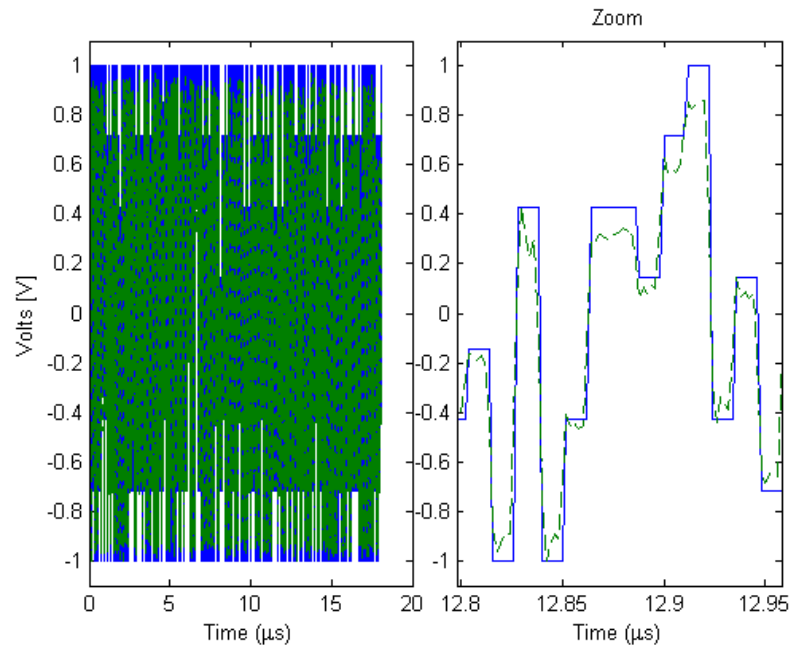


Figura 4.3 Eliminación de ceros x6. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Eliminación de ceros* se cambia la escala de puntos a tiempo y en el zoom se puede observar perfectamente la señal recibida sobre la transmitida. El submuestreo hay dos opciones de hacerlo, como viene explicado en el Procedimiento. Al emplear la “Opción 1: media ponderada” se obtiene la siguiente gráfica.

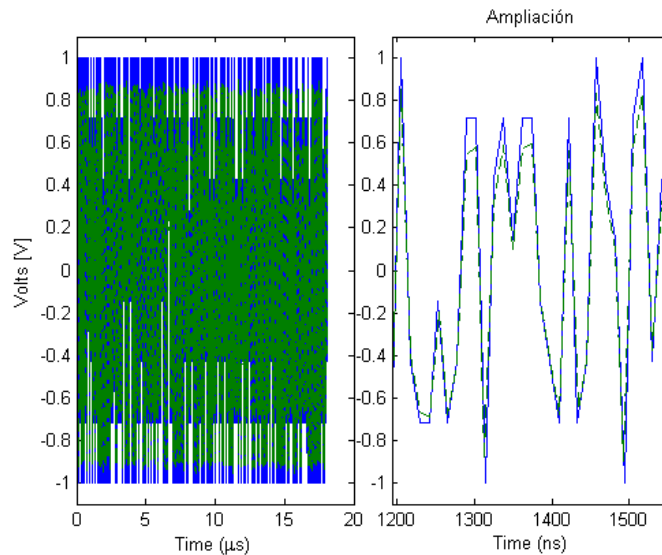


Figura 4.4 Submuestreo media ponderada x6. La Línea sólida corresponde a Tx y la punteada a Rx.

En *Downsampling* (Sub muestreo) de las señales se observa cómo cada bit consta de un solo valor en lugar de 5. Por último, se obtiene el BER con esta opción.

```
Results : BER = 0 | Errors = 0 | Samples = 3225
```

Figura 4.5 BER media ponderada x6

A continuación se procede con la "Opción 2: Interpolación", obteniendo la siguiente gráfica de *Downsampling*.

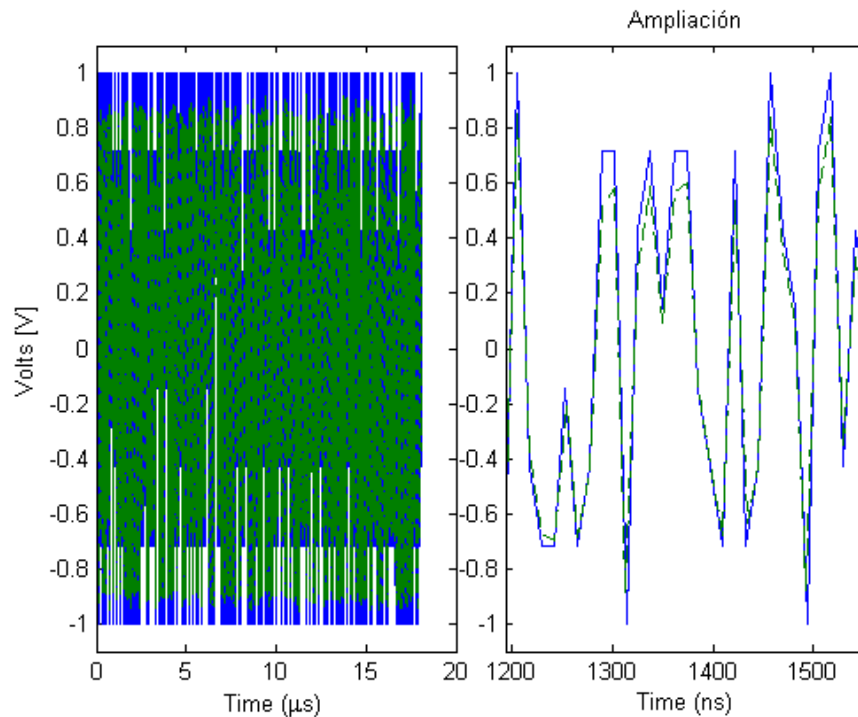


Figura 4.6 Submuestreo interpolación x6. La Línea sólida corresponde a Tx y la punteada a Rx.

Una vez hecho el sub muestreo, se obtiene el BER correspondiente.

```
Results : BER = 0 | Errors = 0 | Samples = 3225
```

Figura 4.7 BER interpolación x6

En este caso de estudio se observan que no hay ningún error en la transmisión, independientemente de método para hacer el sub muestreo empleada. El ancho de bit es de 1 ns con la frecuencia que se estaba buscando: 100 MHz.

4.2. Caso de estudio 2: Portadora a 125 MHz (x5)

En el segundo caso de estudio se un sobremuestreo x5, es decir, que cada bit está compuesto por 5 puntos. Para producir estos datos, se decide que *data* contenga 4.080 datos. De manera que:

$$modData = 4080 * \frac{7}{5} * \frac{1}{3} = 1904$$

Por lo tanto, a la hora de generar los datos, es necesario especificarlo.

```
data = randi([0 1],4080,1);
```

Se procede con el sobremuestreo x5

```
for i = 1 : 1904
    TData((i-1)*5+1:i*5)=modData(i)
end;
TData = TData'
```

En este caso, *TData* contiene 9.520 datos, por lo que se le añaden 480 más para transmitir 10.000 datos.

```
TDataZeros = [zeros(480,1); TData]
```

A continuación se transfieren los datos al generador y se pasan a través del osciloscopio. Para leer los datos correspondientes a este caso, es necesario seleccionar la pestaña correspondiente del archivo.

```
file = 'x5'
```

Una vez seleccionado el caso, se obtiene la *Gráfica con datos de transmisión y señal recibida*.

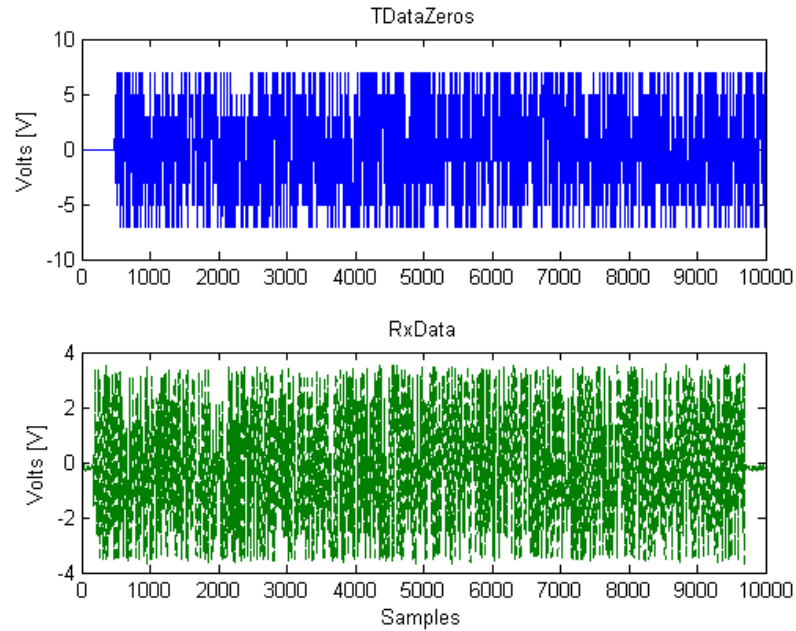


Figura 4.8 Datos de transmisión y señal recibida x5. La Línea sólida corresponde a Tx y la punteada a Rx.

Analizando esta gráfica se observa que ambas señales tienen el mismo número de puntos y periodos, por lo que la frecuencia debe ser la misma. Así mismo, se observa un desfase entre ambas señales, así que se procede a buscar el grupo de ceros para superponerlas.

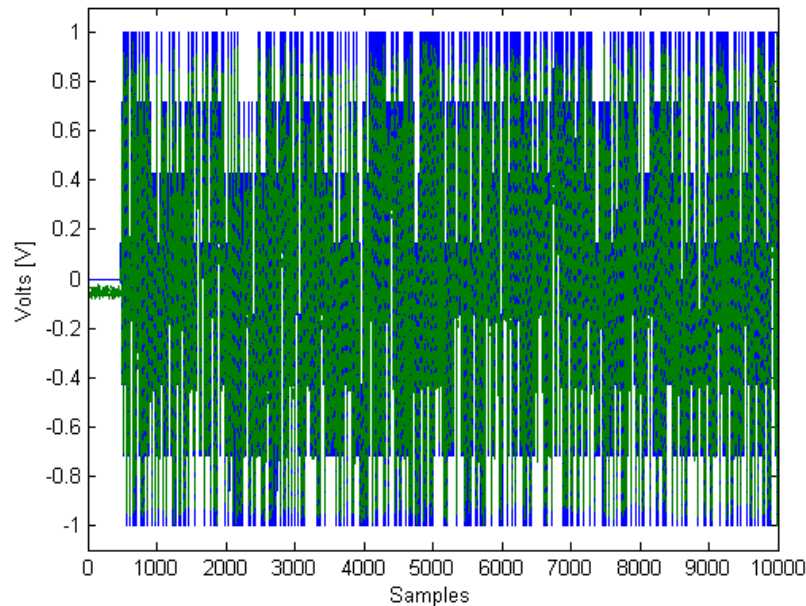


Figura 4.9 Desplazamiento de la señal recibida: correlación x6. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Desplazamiento de la señal recibida: correlación* se observa que hay un número de ceros antes de la que los datos de las señales. El número de ceros es conocido y, en ese caso, es 480. A continuación se procede con la eliminación de los mismos.

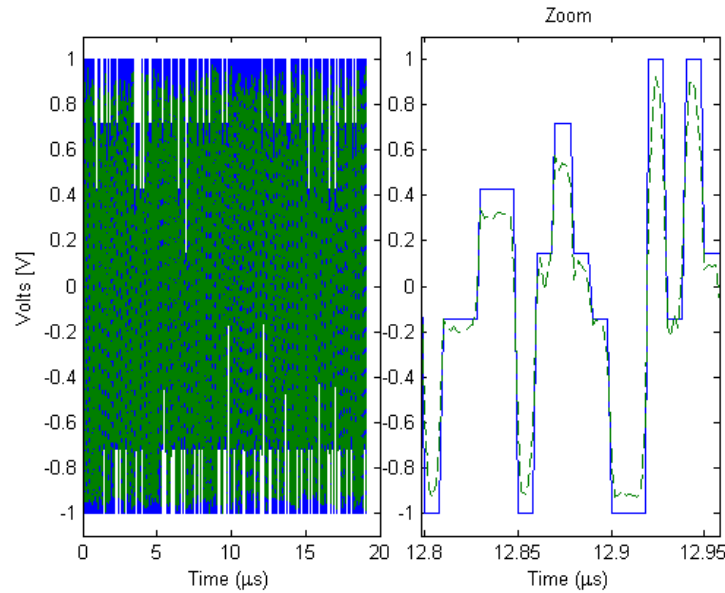


Figura 4.10 Eliminación de ceros x5. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Eliminación de ceros* se cambia la escala de puntos a tiempo y en el zoom se puede observar perfectamente la señal recibida sobre la transmitida. El submuestreo hay dos opciones de hacerlo, como viene explicado en el Procedimiento. Al emplear la “Opción 1: media ponderada” se obtiene la siguiente gráfica.

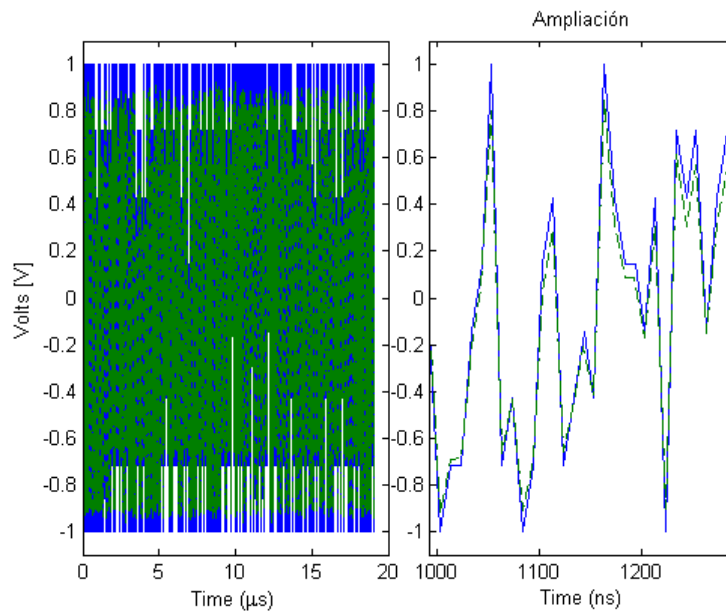


Figura 4.11 Submuestreo media ponderada x5. La Línea sólida corresponde a Tx y la punteada a Rx.

En *Downsampling* (Sub muestreo) de las señales se observa cómo cada bit consta de un solo valor en lugar de 4. Por último, se obtiene el BER con esta opción.

```
Results : BER = 0 | Errors = 0 | Samples = 4080
```

Figura 4.12 BER media ponderada x5

A continuación se procede con la “Opción 2: Interpolación”, obteniendo la siguiente gráfica de *Downsampling*.

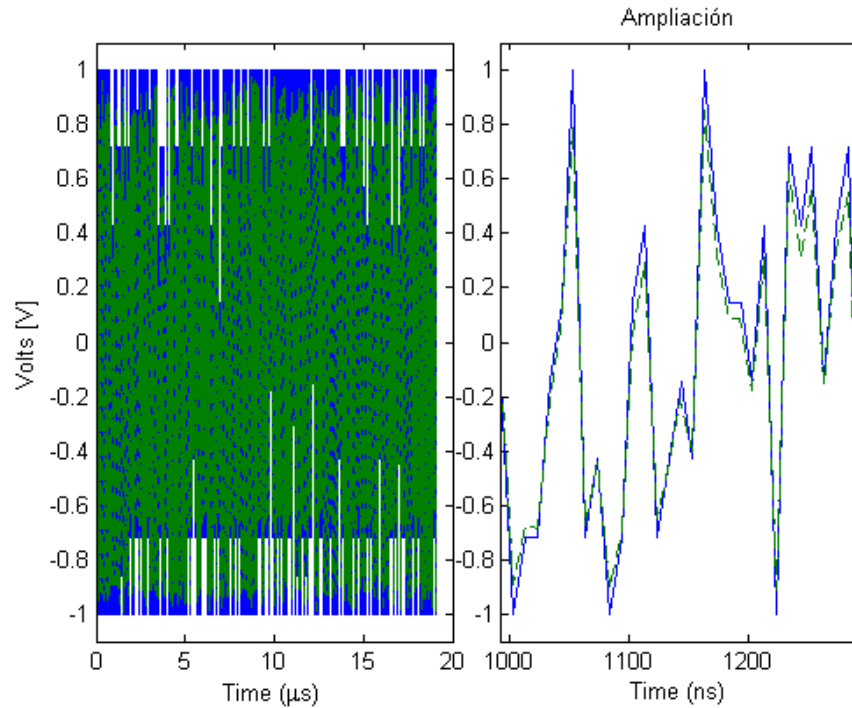


Figura 4.13 Submuestreo interpolación x5. La Línea sólida corresponde a Tx y la punteada a Rx.

Una vez hecho el sub muestreo, se obtiene el BER correspondiente.

```
Results : BER = 0.0034314 | Errors = 14 | Samples = 4080
```

Figura 4.14 BER interpolación x5

En este caso de estudio se observan algunos errores en la transmisión únicamente al emplear la interpolación como método de sub muestreo, ya que con la media ponderada no se producen errores. El ancho de bit es de 8 ns con una frecuencia de 125 MHz.

4.3. Caso de estudio 3: Portadora a 167 MHz (x4)

En el tercer caso de estudio se un sobremuestreo x4, es decir, que cada bit está compuesto por 4 puntos.. Para producir estos datos, se decide que *data* contenga 5.160 datos. De manera que:

$$modData = 5160 * \frac{7}{5} * \frac{1}{3} = 2408$$

Por lo tanto, a la hora de generar los datos, es necesario especificarlo.

```
data = randi([0 1],5160,1);
```

Se procede con el sobremuestreo x4

```
for i = 1 : 2408
    TData((i-1)*4+1:i*4)=modData(i)
end;
TData = TData'
```

En este caso, *TData* contiene 9.632 datos, por lo que se le añaden 368 más para transmitir 10.000 datos.

```
TDataZeros = [zeros(368,1); TData]
```

A continuación se transfieren los datos al generador y se pasan a través del osciloscopio. Para leer los datos correspondientes a este caso, es necesario seleccionar la pestaña correspondiente del archivo.

```
file = 'x4'
```

Una vez seleccionado el caso, se obtiene la *Gráfica con datos de transmisión y señal recibida*.

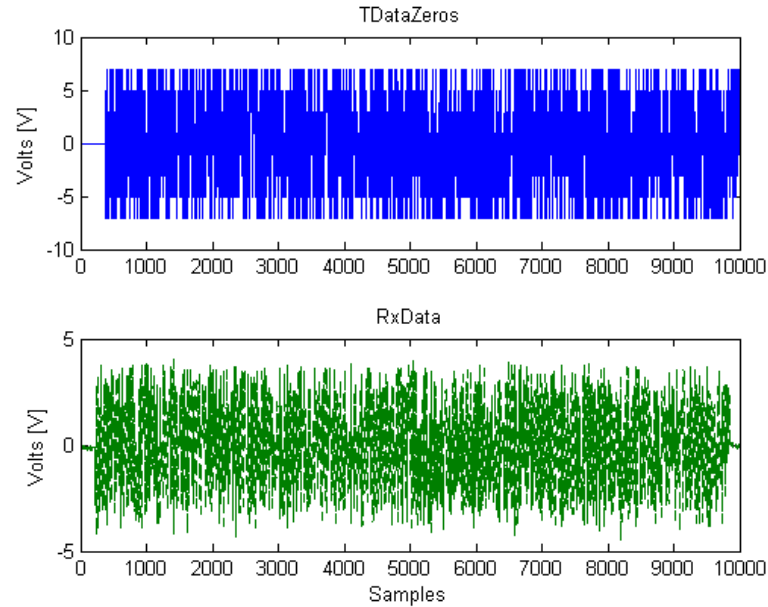


Figura 4.15 Datos de transmisión y señal recibida x4. La Línea sólida corresponde a Tx y la punteada a Rx.

Analizando esta gráfica se observa que ambas señales tienen el mismo número de puntos y periodos, por lo que la frecuencia debe ser la misma. Así mismo, se observa un desfase entre ambas señales, así que se procede a buscar el grupo de ceros para superponerlas.

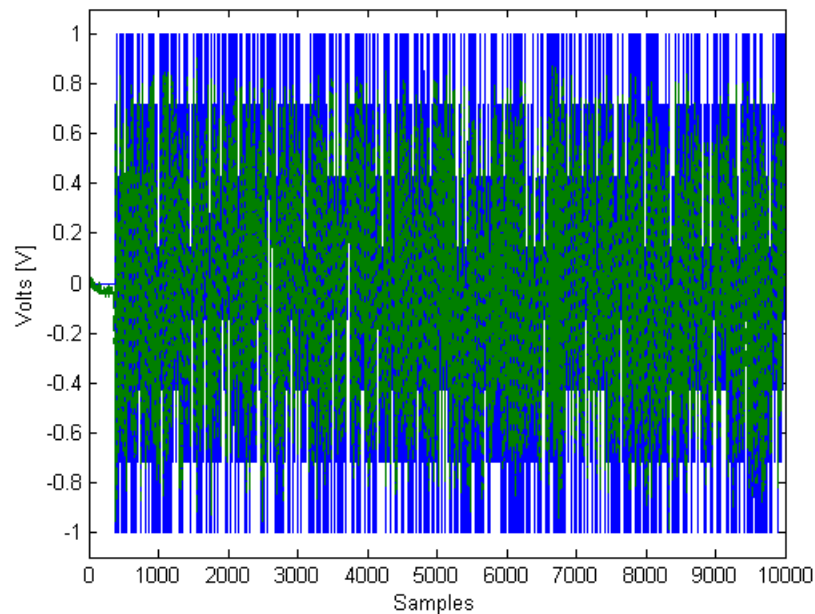


Figura 4.16 Desplazamiento de la señal recibida: correlación x4. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Desplazamiento de la señal recibida: correlación* se observa que hay un número de ceros antes de la que los datos de las señales. El número de ceros es conocido y, en ese caso, es 368. A continuación se procede con la eliminación de los mismos.

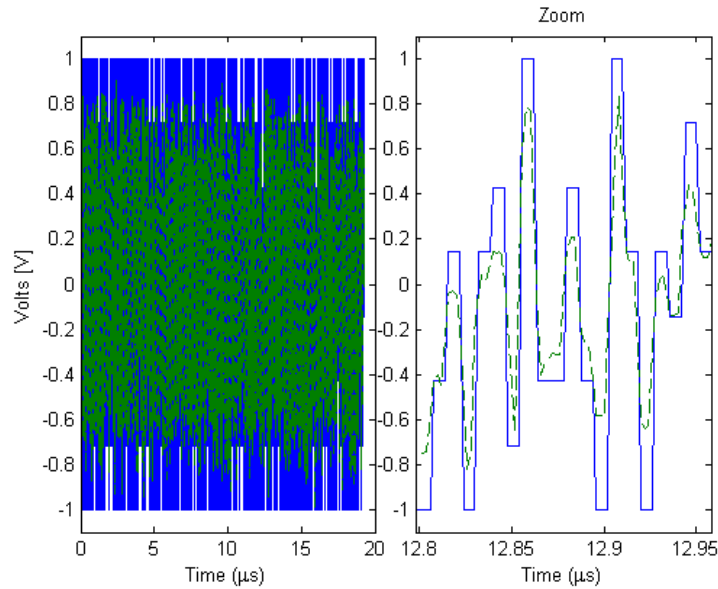


Figura 4.17 Eliminación de ceros x4. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Eliminación de ceros* se cambia la escala de puntos a tiempo y en el zoom se puede observar perfectamente la señal recibida sobre la transmitida. El submuestreo hay dos opciones de hacerlo, como viene explicado en el Procedimiento. Al emplear la “Opción 1: media ponderada” se obtiene la siguiente gráfica.

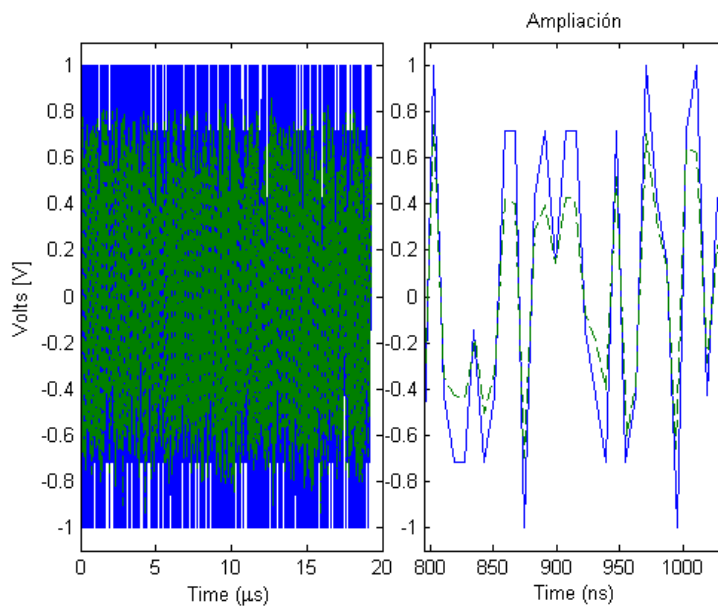


Figura 4.18 Submuestreo media ponderada x4. La Línea sólida corresponde a Tx y la punteada a Rx.

En *Downsampling* (Sub muestreo) de las señales se observa cómo cada bit consta de un solo valor en lugar de 4. Por último, se obtiene el BER con esta opción.

Results : BER = 0.021318 | Errors = 110 | Samples = 5160

Figura 4.19 BER media ponderada x4

A continuación se procede con la "Opción 2: Interpolación", obteniendo la siguiente gráfica de *Downsampling*.

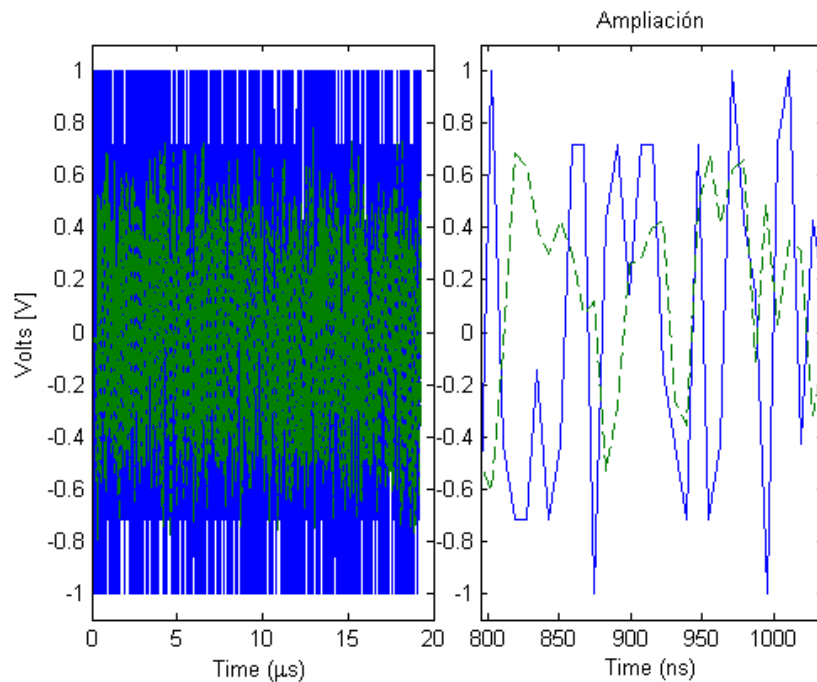


Figura 4.20 Submuestreo interpolación x4. La Línea sólida corresponde a Tx y la punteada a Rx.

Una vez hecho el sub muestreo, se obtiene el BER correspondiente.

Results : BER = 0.16647 | Errors = 859 | Samples = 5160

Figura 4.21 BER interpolación x4

En este caso de estudio se observan algunos errores en la transmisión, sobre todo al usar la interpolación para hacer el sub muestreo. El ancho de bit es de 9 ns con un frecuencia de 166.67 MHz.

4.4. Caso de estudio 4: Portadora a 250 MHz (x3)

En el cuarto caso de estudio se un sobremuestreo x3, es decir, que cada bit está compuesto por 3 puntos. Se trata de un caso límite, ya que la frecuencia de muestreo del osciloscopio es de 500 MHz.

Para producir estos datos, se decide que *data* contenga 6.975 datos. De manera que:

$$modData = 6975 * \frac{7}{5} * \frac{1}{3} = 3255$$

Por lo tanto, a la hora de generar los datos, es necesario especificarlo.

```
data = randi([0 1],6975,1);
```

Se procede con el sobremuestreo x3

```
for i = 1 : 3255
    TData((i-1)*3+1:i*3)=modData(i)
end;
TData = TData'
```

En este caso, *TData* contiene 9.765 datos, por lo que se le añaden 235 más para transmitir 10.000 datos.

```
TDataZeros = [zeros(235,1); TData]
```

A continuación se transfieren los datos al generador y se pasan a través del osciloscopio. Para leer los datos correspondientes a este caso, es necesario seleccionar la pestaña correspondiente del archivo.

```
file = 'x3'
```

Una vez seleccionado el caso, se obtiene la *Gráfica con datos de transmisión y señal recibida*.

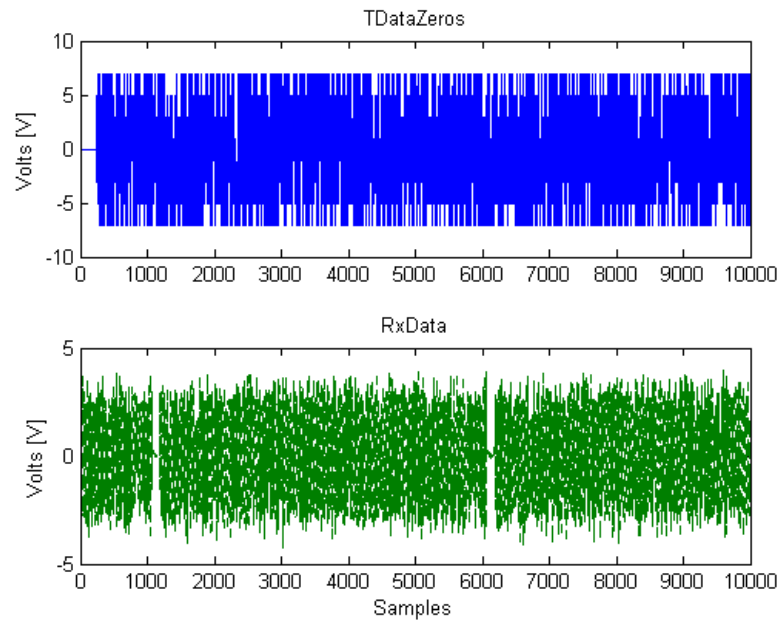


Figura 4.22 Datos de transmisión y señal recibida x3. La Línea sólida corresponde a Tx y la punteada a Rx.

Analizando esta gráfica se observa que ambas señales no tienen el mismo número de puntos y periodos, por lo que la frecuencia de Rx debe ser el doble que la de Tx. Así mismo, se observa un desfase entre ambas señales, así que se procede a buscar el grupo de ceros para superponerlas.

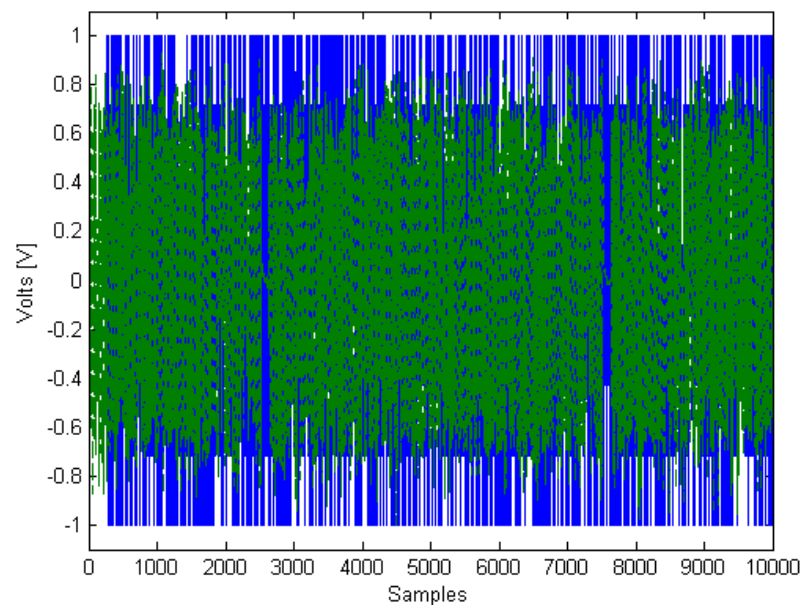


Figura 4.23 Desplazamiento de la señal recibida: correlación x3. La Línea sólida corresponde a Tx y la punteada a Rx.

En la gráfica *Desplazamiento de la señal recibida: correlación* se observa que se trata de un caso límite. En este caso, el procedimiento diseñado no resulta válido, ya que no es posible hacer una correlación válida que superponga las señales correctamente. Ante la imposibilidad de recuperar los datos, se suspende la aplicación del resto del procedimiento.

Capítulo V: Conclusión y posibles mejoras y trabajos futuros

Las siguientes conclusiones son una valoración entre los objetivos propuestos y los resultados finales. En primer lugar, se ha realizado un estado del arte con los últimos experimentos en SI-POF. Por otro lado, también se ha conseguido implementar un sistema de transmisión y detección de señales M-PAM optimizado para sistemas SI-POF, por lo que el objetivo principal se ha cumplido. Además, se ha experimentado con diferentes velocidades, resultando ser el primer caso de estudio (a 100 MHz) el que mejor transmisión ha obtenido. Hay que tener en cuenta, que en este caso, los bits tenían más puntos y, por lo tanto, el osciloscopio ha leído mejor los datos. También me gustaría valorar, la resolución de cambiar de sistema de transmisión una vez que se comprobó que las tarjetas de National Instruments no eran válidas.

Una vez que se implementó el sistema de transmisión con el generador de funciones, este Trabajo sirve de guía para la realización del análisis de la calidad mediante la medición del BER. Simplemente con el hardware especificado y este TFG se podría realizar dicha medición.

Por otro lado, es cierto que se llegó al límite debido a que el osciloscopio sólo permite recuperar 10.000 puntos. La mejor manera de analizar el BER sería con infinitos puntos (irrealizable), por lo que este trabajo se podría mejorar con otro hardware que permitiera analizar muchos más puntos que los que se han analizado en este proyecto. Una futura mejora sería no tener esta limitación de hardware y poder analizar muchos más bits en este tipo de señales. Una posible mejora sería con un osciloscopio que permitiese recuperar más datos. Otra posible mejora sería con unas tarjetas de National Instruments más potentes que sí permitiesen la transmisión a la velocidad deseada.

Además, en este caso, la transmisión ha sido mediante un cable coaxial de 1 metro de longitud. Un posible trabajo futuro podría ser implementar este procedimiento sobre un enlace SI-POF.

Es importante destacar que el objetivo de este trabajo es demostrar el procedimiento básico de la transmisión de señales M-PAM para su posterior análisis en términos de BER. Por ejemplo, si se desea obtener un sistema totalmente automatizado, se podría interconectar el trigger del generador de funciones con el del osciloscopio, y leer los datos del osciloscopio a través de un cable RS232.

Gracias a este Trabajo de Fin de Grado he aprendido cómo se realiza un proyecto correctamente, en cuanto a obtención de información, planificación, experimentación y redacción. Hay que resaltar la importancia de una buena comunicación dentro de un equipo, como en este caso.

Anexo A: Presupuesto

Este capítulo está dedicado a los costes económicos que tiene la realización de este proyecto. Para ello, los costes se han dividido en equipos de medida, software y trabajo de ingeniería. En la última tabla se puede observar el coste total con IVA.

Para la realización de este Trabajo de Fin de Grado no ha sido necesaria la adquisición de ningún elemento. El presupuesto total de este trabajo está dividido en costes de software y equipos de medida, y trabajo de ingeniería.

2.1. Costes de software y equipos de medida

Aunque los equipos de medida empleados en la implantación de este Trabajo de Fin de Grado son propiedad de la Universidad Carlos III, hay que tener en cuenta su coste de no haber realizado el proyecto en la universidad. Debido a la antigüedad del Generador empleado se ha buscado un equivalente en el mercado.

Descripción del equipo	Precio de lista
Tektronix TDS3052	7.362,00 €
Generador HP 33502A	1.182,00 €
MATLAB	446.85 €
Coste A	8.990,85 €

Tabla 5 Coste de software y equipos de medida

Debido a que no se trata de un proyecto de larga duración, se podría optar por la compra de segunda mano de los equipos o por el alquiler de los mismos por las semanas que fueran necesarios por las semanas que fueran necesarias.

2.2. Trabajo de ingeniería

Teniendo en cuenta la mano de obra del ingeniero junior en base al mercado laboral se puede estimar el siguiente coste.

Descripción de la actividad	Precio por hora	Horas empleadas	Coste total
Investigación modulación	35 €	25	875.00 €
Conocimiento MATLAB	35 €	35	1.225,00 €
Desarrollo sistemas	35 €	65	2.275,00 €
Implementación	35 €	55	1.925,00 €
Redacción memoria	15 €	120	1.800,00 €
Coste B			8.100,00 €

Tabla 6 Coste del trabajo de ingeniería

2.3. Coste total

La siguiente tabla muestra el coste total de este Trabajo de Fin de Grado.

Descripción	Coste
Coste A	8.990,85 €
Coste B	8.100,00 €
EBITDA (A + B)	17.090,85 €
IVA (21 %)	3.589,08 €
COSTE TOTAL	20.679,93 €

Tabla 7 Coste total del TFG

Por lo tanto, como se puede observar en la tabla anterior el coste de la implantación de un sistemas de transmisión y detección de señales M-PAM sobre Fibra óptica de plástico tiene un coste total aproximado de 20.679,93 €.

Anexo B: Código figuras Matlab

En este Anexo, se va exponer el código de matlab necesario para crear las figuras en los casos de estudio.

Gráfica con datos de transmisión y señal recibida

```
figure('Name', 'Gráfica con datos de transmisión y señal recibida')

subplot(2,1,1)
    x = 1:length(Tx0);
    plot(x,Tx0, 'b');
    ylabel('Volts [V]');
    title('TDataZeros')

subplot(2,1,2)
    x = 1:length(Rx0);
    plot(x,Rx0,'--', 'Color', [0 0.5 0]);
    xlabel('Samples');
    ylabel('Volts [V]');
    title('RxData')
```

Desplazamiento de la señal recibida: correlación

```
figure('Name', 'Desplazamiento de la señal recibida: Correlación')
    plot(TxC); hold on; plot(RxC, '--', 'Color', [0 0.5 0]);
    xlabel('Samples');
    ylabel('Volts [V]');
    ylim([-1.1 1.1])
```

Eliminación de ceros iniciales

```
figure('Name', 'Eliminación de ceros iniciales')
subplot(1,4,1:2)
plot(timeTx/1e-6, DataTx, timeRx/1e-6, DataRx, '--')
    xlabel('Time (\mus)');
    ylabel('Volts [V]');
    ylim([-1.1 1.1])
subplot(1,4,3:4)
plot(timeTx(6400:6480)/1e-6, DataTx(6400:6480), timeRx(6400:6480)/1e-6,
DataRx(6400:6480), '--')
    xlabel('Time (\mus)');
    title('Zoom');
    ylim([-1.1 1.1]); xlim([timeTx(6400), timeTx(6480)]/1e-6)
```


Downsampling (sub muestreo) de las señales

```
f = figure('Name', 'Downsampling (Sub Muestreo) de las señales');

time = 0:length(DTx)-1;
subplot(1,4,1:2)
plot(tTx/1e-6, DTx, tRx/1e-6, DRx, '--');
    xlabel('Time (\mus)');
    ylabel('Volts [V]');
    ylim([-1.1 1.1])
subplot(1,4,3:4)
plot(tTx(100:130)/1e-9, DTx(100:130), tRx(100:130)/1e-9, DRx(100:130), '--');
    xlabel('Time (ns)');
    title('Ampliación')
    ylim([-1.1 1.1]); xlim([tTx(100) tTx(130)]/1e-9)
```

Anexo C: Referencias

- [1] <http://www.kdpof.com/technology/what-is-pof/>
- [2] O. Ziemman, J. Krauser, P. E. Zamzow, and W. Daum, POF Handbook: Optical Short Range Transmission Systems. New York, NY, USA: Springer-Verlag, 2008
- [3] M. Atef, R. Swoboda, and H. Zimmermann, "A gigabit fully integrated plastic optical fiber receiver for a RC-LED source," in Proc. IEEE 15th Des. Diagn. Electro. Circuits Syst. Int. Symp., Apr. 18–20, 2012, pp. 74–78.
- [4] http://www.hs-schmalkalden.de/schmalkaldenmedia/WDM_Key_Element_in_the_visible_spectrum_for_Standard_SI_POF-p-13322.pdf
- [5] A. M. J. Koonen, and E. Tangdionga, "Photonic Home Area Networks", Journal of Lightwave Technology, 32(4), 591-604, 2014.
- [6] Kragl, H., "A product family for simplex POF home networks", http://www.pofac.de/itg/fg_5_4_1/de/fachgruppentreffen/treffen21.php, (2006)
- [7] Chen, R. T. and Lipscomb, G. F., "WDM and Photonic Switching Devices for Network Applications", Proceedings of SPIE 3949, (2000)
- [8] POF Handbook: Optical Short Range Transmission Systems Escrito por Olaf Ziemann, Jürgen Krauser, Peter E. Zamzow, Werner Daum. página 80-81
- [9] J.L. Wei, L. Geng, D.G. Cunningham, R.V. Penty, and I.H. White, "Gigabit NRZ, CAP and optical OFDM systems over POF links using LEDs," Opt. Express 20, 2284–22289, 2012.
- [10] S Loquai, R Kruglov, B Schmauss, C.-A. Bunge, F Winkler, O Ziemann, E Hartl, and T Kupfer "Comparison of Modulation Schemes for 10.7 Gb/s Transmission Over Large-Core 1 mm PMMA Polymer Optical Fiber" Journal of Lightwave Technology, 31 (13), 2170-2176, 2013.
- [11] "Tipos de modulación" <http://telefoliomodulacion.blogspot.com.es/2009/10/tipos-de-modulacion.html>
- [12] Amir H. Abdolhamid, David A. Johns "A comparison of CAP/QAM architectures" Universidad de Toronto
- [13] Daum, W. and Krauser, J. and Zamzow, P. E. and Ziemann, O., "POF – Polymer Optical Fibers for Data Communication", Springer-Verlag, Heidelberg (2002)
- [14] Nalwa, H. S., "Polymer Optical Fibers", American Scientific Publishers, California, (2004)

- [15] A. Antonino, S. Straullu, S. Abrate, A. Nespola, P. Savio, D. Zeolla, Julio R. Molina, R. Gaudino, Sven Loquai and Juri Vinogradov "Real-time Gigabit Ethernet bidirectional transmission over a single SI-POF up to 75 meters " (2010)
- [16] X. Li, J. L. Wei, N. Bamiedakis, R. V. Penty, and I. H. White, "Avalanche photodiode enhanced PAM-32 5 Gb/s LED-POF link," presented at the Eur. Conf. Opt. Commun., Cannes, France, Sep. 21–25, 2014.
- [17] J. Vinogradov, R. Kruglov, K. L. Chi, J. W. Shi, M. Bloos, S. Loquai, and O. Ziemann, "GaN light-emitting diodes for up to 5.5-Gb/s short-reach data transmission over SI-POF," IEEE Photon. Technol. Lett., vol. 26, no. 24, pp. 2473–2475, Dec. 15, 2014.
- [18] Bamiedakis, N., Li, X., Ferreira, R., Meng, X., McKendry, J. J. D., Xie, E., Gu, E., et al. (2016). Research data supporting "8 Gb/s PAM-16 data transmission over 10 m of SI POF using novel 450 nm micro-pixelated LEDs"
- [19] X. Li, N. Bamiedakis, J. L. Wei, J. J. D. McKendry, E. Xie, R. Ferreira, E. Gu, M. D. Dawson, R. V. Penty, and I. H. White, "6.25 Gb/s POF link using GaN μ LED arrays and optically generated pulse amplitude modulation," presented at the Conf. Lasers Electro.-Opt., San Jose, CA, USA, May 10–15, 2015.
- [20] Xin Li, Nikolaos Bamiedakis, Jinlong Wei, Jonathan J. D. McKendry, Enyuan Xie, Ricardo Ferreira, Erdan Gu, Martin D. Dawson, Richard V. Penty, and Ian H. White " μ LED-Based Single-Wavelength Bi-directional POF Link With 10 Gb/s Aggregate Data Rate" (2015)
- [21] U. H. P. Fischer, M. Haupt and M. Joncic, "Optical Transmission Systems Using Polymeric Fibers", (InTech, 2011).
- [22] M. Joncic, M. Haupt, U.H.P. Fischer, "Standardization Proposal for Spectral Grid for VIS WDM Applications over SI-POF", in Proceedings of POF Congress, 351-355, 2012.
- [23] R. Kruglov, et al., "10.7-Gbit/s Discrete Multitone Transmission over 50-m SI-POF Based on WDM Technology," IEEE Photon. Technol. Lett., vol. 24, no. 18, 1632-1634 (2012).
- [24] R. Kruglov, J. Vinogradov, S. Loquai, O. Ziemann, C. A. Bunge, T. Hager, and U. Strauss, "21.4 Gb/s discrete multitone transmission over 50-m SIPOF employing 6-channel WDM," presented at the Opt. Fiber Commun. Conf. Exhib., San Francisco, CA, USA, Mar. 9–13, 2014.
- [25] R. Caspary, et al., "High Speed WDM Transmission on Standard Polymer Optical Fiber," in Proc. ICTON 2015, (Budapest, 2015) Tu.B1.2.
- [26] Xin Li, Nikolaos Bamiedakis, Jonathan Mckendry, Enyuan Xie, Ricardo Ferreira, Erdan Gu, Martin Dawson, Richard Penty, and Ian H. White "11 Gb/s WDM Transmission Over SI-POF Using Violet, Blue and Green μ LEDs" presented at the Optical Fiber Communications Conference (2016)

[27] R. Kruglov¹ , S. Loquai¹ , J. Vinogradov¹ , O. Ziemann¹ , C.-A. Bunge² , G. Bruederl³ , U. Strauss³ "10.7 Gb/s WDM Transmission over 100-m SI-POF with Discrete Multitone" (2016)